UNIVERSIDADE FEDERAL DO PARANÁ
DEPARTAMENTO DE CIÊNCIA E GESTÃO DA INFORMAÇÃO

HELTON YUKIO HATORI

SISTEMA ONLINE PARA MINERAÇÃO DE DADOS: MÉTODOS C4.5 E APRIORI

CURITIBA
2012

HELTON YUKIO HATORI

SISTEMA ONLINE PARA MINERAÇÃO DE DADOS: MÉTODOS C4.5 E APRIORI

Monografia apresentada à disciplina Pesquisa em Informação II como requisito parcial à conclusão do Curso de Gestão da Informação, do Departamento de Ciência e Gestão da Informação, do Setor de Ciências Sociais Aplicadas da Universidade Federal do Paraná.

Orientadora: Profª. Dra. Denise Fukumi Tsunoda

CURITIBA
2012

RESUMO

Elabora-se uma ferramenta para Mineração de Dados para aplicação dos algoritmos C4.5 e Apriori. Tem por objetivo que essa ferramenta apresente as seguintes características: aplicação em ambiente web, demonstração dos cálculos e etapas detalhadas dos algoritmos, e apresentação de resultados satisfatórios, em comparação aos sistemas existentes na área. A metodologia utilizada para realização procedeu da seguinte forma: inicialmente foram definidos conceitos pertinentes à pesquisa, sendo eles Dado, Informação e Conhecimento, Bancos de Dados, *Knowledge Discovery in Databases*, Mineração de Dados, Heurísticas e Linguagem de Programação *Web.* Na sequência foram definidas e comparadas características de ferramentas já existentes e, a partir disso, estabelecidas as especificações do sistema. Após isso, foram desenvolvidos os algoritmos em códigos de linguagem HTML e PHP e implantados em um servidor web. Finalmente, aplicou-se o mesmo grupo de dados para as ferramentas Weka, RapidMiner, Tanagra, R e Orange, de forma a comparar os resultados apresentados por todos os aplicativos, inclusive o proposto. Os resultados obtidos ao final da pesquisa se mostraram satisfatórios, visto que os objetivos apresentados foram atingidos.

Palavras-chave: Mineração de Dados; C4.5; Apriori;

ABSTRACT

This paper presents a tool for Data Mining algorithms for application of C4.5 and Apriori. This tool presents the following characteristics: application on web environment, calculus demonstration and detailed steps of the algorithms, and the presentation of satisfactory results in comparison to the current systems on the area. The methodology used to this realization was held following the sequence: first were defined intrinsic concepts to the research, being those Data, Information and Knowledge, Database, Knowledge Discovery in Databases, Data Mining, Heuristics and Web Programming Language. Later were defined and compared current characteristics tools and, starting from this point, established the system specifications. Then were developed the algorithms in HTML and HPH language codes and implanted in a web server. Finally the same group was applied to the Weka, RapidMiner, Tanagra, R and Orange tools, thus comparing the showed results for all the applications, including the proposed one. The obtained results on the research at its final were satisfactory, as the showed goals were achieved.

Key words: Data Mining; C4.5; Apriori

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE SIGLAS

KDD	*Knowledge Discovery in Databases*
DCBD	Descoberta de Conhecimento em Base de Dados
MD	Mineração de Dados
BDO	Banco de Dados Operacionais
DW	*Data Warehouse*
DM	*Data Mart*
ID3	*Iterative Dichotomizer 3*
WEKA	*Waikato Environment Knowledge Analysis*
MySQL	*My Structured Query Language*
SGBD	Sistema Gerenciador de Banco de Dados

SUMÁRIO

1 INTRODUÇÃO

Com o aumento do volume de dados na era da informação e a facilidade de armazenamento, tornou-se imprescindível a análise e recuperação de informação útil em meio à saturação de dados sem importância. É partindo desse cenário que surge a Mineração de Dados.

A área da Mineração de Dados surgiu no final da década de 80 e vem se tornando uma das principais práticas para análise de dados. Para tanto, são utilizados diversos algoritmos, cada qual objetivando um modo de descobrir conhecimento.

Para a aplicação das técnicas da Mineração, são muitas ferramentas existentes, que diferem na quantidade de algoritmos disponíveis, no preço da licença, entre outros aspectos, englobados pela usabilidade, desempenho e fatores ergonômicos.

Dessa forma, o presente documento tem como finalidade registrar a concepção de um sistema para aplicação da Mineração de Dados, especificamente dos algoritmos C4.5 e Apriori, baseando-se em softwares existentes com características semelhantes, mas que permita a utilização em um ambiente *web*.

1.1 Tema

Elaboração de um sistema online para mineração de dados utilizando os algoritmos C4.5 e Apriori.

1.2 Problema de Pesquisa

Cada vez mais, a Mineração de Dados se torna necessária para a gestão da informação, visto que a quantidade de dados armazenados em bases de dados é enorme e continua em constante crescimento, consequência da queda dos custos para armazenamento dos dados. Devido a tais fatores, somado à incapacidade do ser humano de interpretar tamanha quantidade de dados, tem-se a necessidade de criação de novas ferramentas e técnicas de extração do conhecimento. (REZENDE, 2005)

No entanto, as ferramentas existentes, quando gratuitas, requerem do usuário a instalação em máquina local e atualizações do programa, custando tempo do download e instalação, uso de memória e espaço em disco, além de ferramentas que obrigam o usuário a ter um determinado sistema operacional.

A partir de tal panorama, pode-se chegar a um questionamento: **como elaborar uma ferramenta para mineração de dados que possa ser acessada via internet, sem necessidade de instalação, que apresente as etapas do processamento e resultados satisfatórios, em comparação a cinco softwares da área, mais utilizados em ambientes acadêmicos?**

1.3 Justificativa

Visto que atualmente não existe uma ferramenta online na área da mineração de dados, faz-se obrigatória a instalação em máquina local de qualquer software que se queira utilizar. O presente trabalho apresenta como resultado uma opção alternativa aos softwares existentes, possuindo toda sua interface em ambiente *web*.

Do ponto de vista didático, o trabalho resultará em uma ferramenta a ser utilizada nas disciplinas que abordam a Mineração de Dados. Para o docente, o sistema apresenta em detalhes os métodos e os resultados exibidos nas diferentes heurísticas. Para o discente, viabiliza a visualização da prática e compreensão do conteúdo ministrado. Finalmente, tanto para o professor quanto para os alunos ainda há a possibilidade de uso do sistema para conferência de resultados dos exercícios propostos/apresentados em sala de aula.

Aos interessados na área de Mineração de Dados, o software viabiliza uma forma prática de obtenção de informações de base de dados, sem a necessidade de download e instalação da ferramenta *in loco*, necessitando somente de um acesso à internet. Pela característica de possuir um código aberto, ainda poderá ser incorporado em outros sistemas online para atender as necessidades do usuário, além de possibilitar que outros métodos sejam adicionados.

Para o autor, o tema se faz relevante por agregar conhecimento na área de programação *web* e na área de Mineração de Dados, mostrando as possibilidades da análise da informação em repositórios de dados.

O trabalho, por fim, apresenta-se como uma iniciativa, que poderá ser continuada com a aplicação de outros métodos existentes na Mineração de Dados, em se havendo o interesse ou a necessidade.

1.4 Objetivos

Como objetivo geral define-se, elaborar uma ferramenta online que possibilite a mineração de dados externos retornando o processo detalhado dos algoritmos C4.5 e Apriori, assim como resultados potencialmente relevantes para o usuário.

Partindo do objetivo geral foi possível estabelecer os seguintes objetivos específicos:

a) estudar os principais conceitos relacionados à pesquisa;
b) definir critérios para análise dos software gratuitos selecionados;
c) estabelecer características da ferramenta;
d) implementar o sistema em um servidor *web*;
e) analisar os resultados.

1.5 Estrutura do trabalho

Primeiramente é feita uma abordagem dos principais aspectos teóricos ligados à MD. Define-se dado, informação e conhecimento, bases de dados (*Data Warehouse*, *Data Mart* e Bancos de Dados Operacionais), KDD, suas etapas e características, e no âmbito da MD são explicadas as tarefas, heurísticas e algoritmos C4.5 e Apriori (explicados na seção 2). Em seguida são descritos as linguagens de programação utilizadas para elaboração do sistema.

Na seção 3 são descritas as características, métodos e limitações da pesquisa. Conta também com a definição de critérios e a comparação entre ferramentas para MD.

Na seção 4, descrevem-se as especificações do sistema desenvolvido, explica-se sua utilização, apresentação, funcionamento, análise dos resultados obtidos pelo sistema e comparativo da ferramenta elaborada com as citadas na seção 3.

Na sequência é feita a avaliação dos resultados da pesquisa, comparando os resultados obtidos pelo sistema desenvolvido com os resultados das outras ferramentas já existentes. Por fim são feitas as considerações finais, baseadas na revisão de literatura, na avaliação das ferramentas de MD, no desenvolvimento do sistema online e na avaliação dos resultados obtidos.

2 REFERENCIAL TEÓRICO

Para entender do que se trata o sistema desenvolvido, é necessário um embasamento teórico, em literaturas pertinentes, acerca dos conceitos relacionados à mineração de dados e às estruturas utilizadas para o funcionamento do sistema.

2.1 Dado, informação e conhecimento

Levando-se em consideração que a mineração de dados trabalha diretamente com a análise de dados brutos para a busca por padrões informacionais e extração de conhecimento, faz-se necessária a conceituação e diferenciação entre dado, informação e conhecimento.

Setzer (1999, p.2) define dado como uma “sequência de símbolos quantificados ou quantificáveis” que podem ser armazenados e processados em um computador. Como exemplos de dados podem ser citados imagens, textos, sons e animação.

De forma análoga, Davenport (1998, p.18) conceitua dados como sendo “simples observações sobre o estado do mundo”, caracterizando-os como facilmente estruturados, facilmente obtidos por máquinas, frequentemente quantificados e de fácil transferência.

De acordo com Miranda (1999, p.286) “dado é o conjunto de registros qualitativos ou quantitativos conhecido que organizado, agrupado, categorizado e padronizado adequadamente transforma-se em informação”. Essa definição apresenta a relação mais comum entre dado e informação vista em literaturas da área. Ao definir informação, o autor ainda faz referência à importância da informação como subsídio para a tomada de decisão.

Davenport (1998, p.19) cita Peter Drucker e sua definição de informação como sendo “dados dotados de relevância e propósito” complementando que é a mediação humana que fornece tais atributos aos dados tornando difícil sua transferência com absoluta fidelidade. Setzer (1999, p.2) considera que essa interpretação humana é o que diferencia dado, totalmente sintático, de informação, em parte semântica.

O conhecimento, por sua vez, envolve uma abstração pessoal sobre uma informação já experimentada. Nesse contexto, o conhecimento não pode ser

inserido em um computador, pois seria reduzido a uma simples informação (SETZER, 1999). Miranda (1999, p.287), por outro lado, apresenta três tipos diferentes de conhecimento: explícito, tácito e estratégico. O conhecimento explícito é “o conjunto de informações elicitadas em algum suporte”, o tácito é “o acúmulo de saber prático sobre um determinado assunto” agregado a fatores da personalidade da pessoa, e o estratégico é a combinação dos conhecimentos explícito e tácito.

A definição de conhecimento explícito para Miranda (1999) se assemelha à caracterização de informação de Setzer (1999), ao mesmo tempo em que o conhecimento tácito conceituado se encontra parelho ao conhecimento do mesmo autor.

Para Davenport (1998, p.19) conhecimento é a informação mais valiosa da mente humana, posta sob um contexto, um significado, uma interpretação. Por muitas vezes é tácito, mas pode ser incorporado em máquinas, mesmo que seja de difícil categorização e localização.

As definições apresentadas mostram que são raras ‘bases de informações’ ou ‘bases de conhecimento’ pelo fato de que tanto informações quanto conhecimentos são difíceis de serem armazenados. Portanto, têm-se bancos de dados, dos quais são possíveis de se obter informações e até conhecimento.

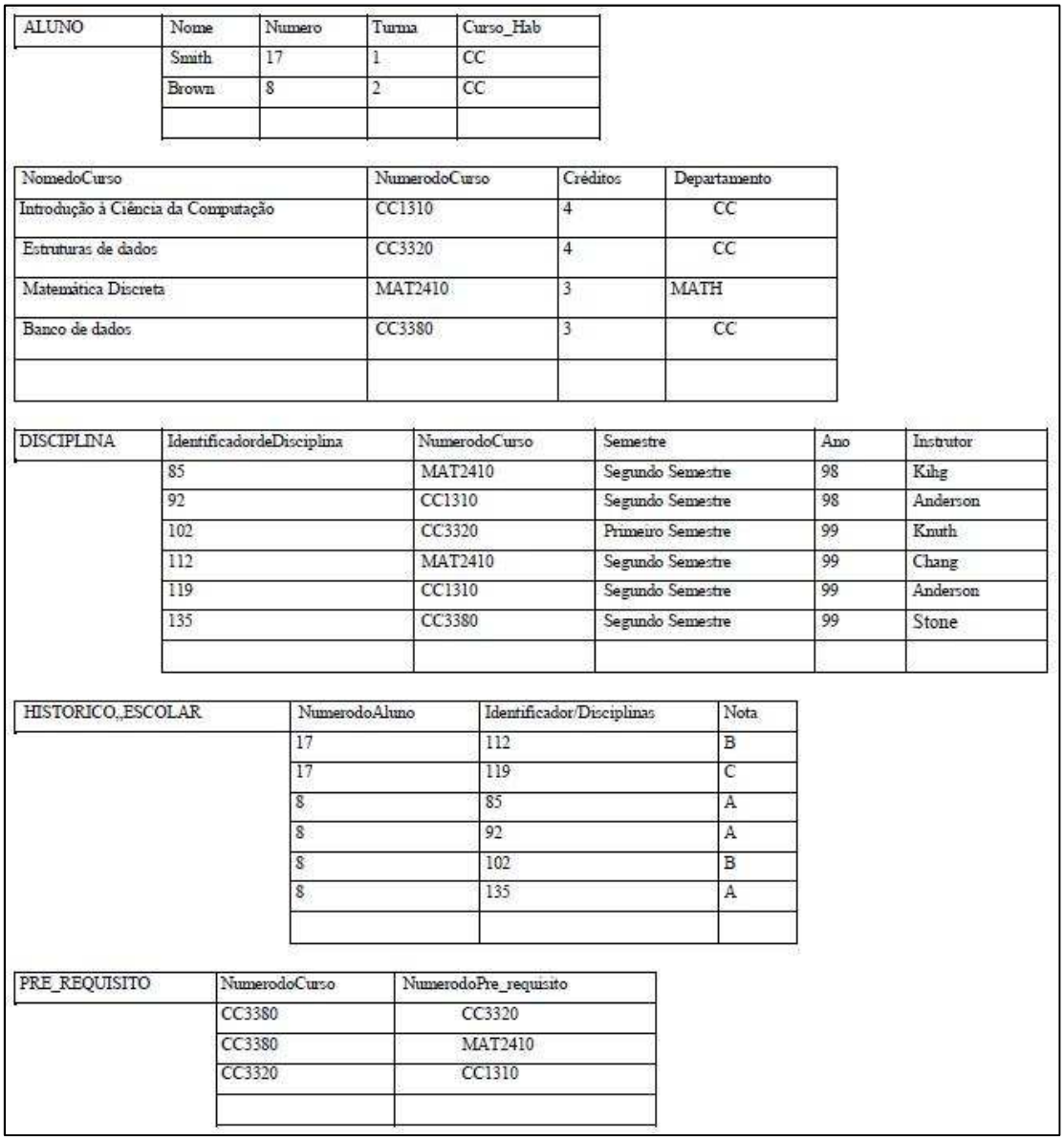
2.2 Bancos de dados

Esta seção discorre sobre três estruturas de bancos de dados, a saber: *Data Warehouse*, *Data Mart* e Banco de Dados Operacionais.

2.2.1 *Data Warehouse*

Quando o assunto é *Data Warehouse* (DW), existem duas vertentes principais, criadas por Willian Inmon (1997) e por Ralph Kimball (2002), que possuem diferentes filosofias, técnicas de modelagem e estratégias de implementação. Para Inmon (1997), o *Data Warehouse* é definido como uma coleção de dados integrada, consistente, variável em relação ao tempo, utilizadas principalmente para apoio ao processo de tomada de decisões administrativas. Essencialmente é uma base de dados que agrega informações extraídas de

múltiplas fontes de informação menores, criada a partir de um modelo relacional, ou seja, elaborada como uma coleção de relações entre as tabelas de valores como pode ser visto na Figura 1, onde os registros da tabela ALUNO estão relacionados à tabela HISTÓRICO ESCOLAR por meio do atributo 'NumerodoAluno', que por sua vez faz uma ligação com a tabela DISCIPLINA, através do atributo 'IdentificadordeDisciplina', e que interliga os dados das tabelas restantes através do 'NúmerodoCurso'.



ALUNO	Nome	Numero	Turma	Curso_Hab
	Smith	17	1	CC
	Brown	8	2	CC

NomedoCurso	NumerodoCurso	Créditos	Departamento
Introdução à Ciência da Computação	CC1310	4	CC
Estruturas de dados	CC3320	4	CC
Matemática Discreta	MAT2410	3	MATH
Banco de dados	CC3380	3	CC

DISCIPLINA	IdentificadordeDisciplina	NumerodoCurso	Semestre	Ano	Instrutor
	85	MAT2410	Segundo Semestre	98	Kihg
	92	CC1310	Segundo Semestre	98	Anderson
	102	CC3320	Primeiro Semestre	99	Knuth
	112	MAT2410	Segundo Semestre	99	Chang
	119	CC1310	Segundo Semestre	99	Anderson
	135	CC3380	Segundo Semestre	99	Stone

HISTORICO„ESCOLAR	NumerodoAluno	Identificador/Disciplinas	Nota
	17	112	B
	17	119	C
	8	85	A
	8	92	A
	8	102	B
	8	135	A

PRE_REQUISITO	NumerodoCurso	NumerodoPre_requisito
	CC3380	CC3320
	CC3380	MAT2410
	CC3320	CC1310

Figura 1 - Exemplo de tabelas em um modelo relacional
Fonte: ELMASRI e NAVATHE (2005, p.5)

A definição dada por Kimball (2002) explica que é o conglomerado das fontes de informações menores que formam um DW. Para o autor citado, deve-se

estruturar os dados em um modelo multidimensional, podendo esse ser um modelo Estrela (Star) ou um modelo Floco de Neve (Snow Flake).

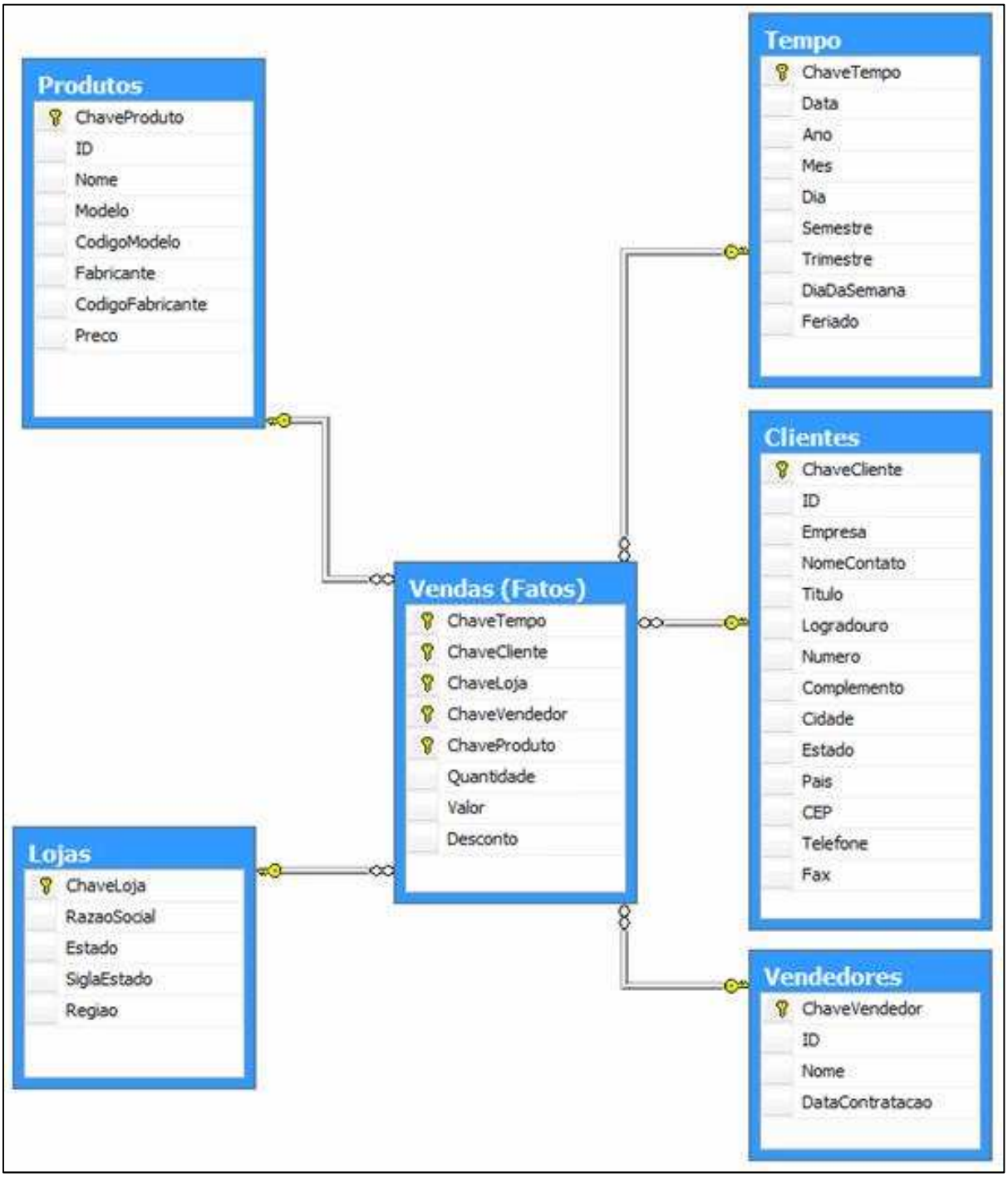


Figura 2 - Representação de um modelo estrela
Fonte: NARDI (2007)

Nardi (2007) descreve que a finalidade de bases de dados multidimensionais é fornecer subsídio para realização de análises, utilizando-se do cruzamento de dados. Para esses modelos, é necessária a existência de uma tabela com os dados

principais a serem agrupados, chamados de fatos, e tabelas que derivam dos fatos, possibilitando a organização dos dados, chamados de dimensões. No modelo Estrela, a tabela de fatos se encontra centralizada e as dimensões se encontram ao redor, como apresentado na Figura 2. Nardi (2007) ainda descreve que esse modelo caracteriza-se pela simplicidade e eficiência, facilitando a definição de hierarquias, mas que, no entanto, não apresenta normalização nas tabelas de dimensões. Já o Floco de Neve, visto na Figura 3, é um modelo Estrela com o acréscimo de normalização nas tabelas de dimensões, eliminando redundâncias que tornam ágeis as manutenções nas tabelas. Entretanto, pela maior quantidade de tabelas em junções, pode haver queda de desempenho.

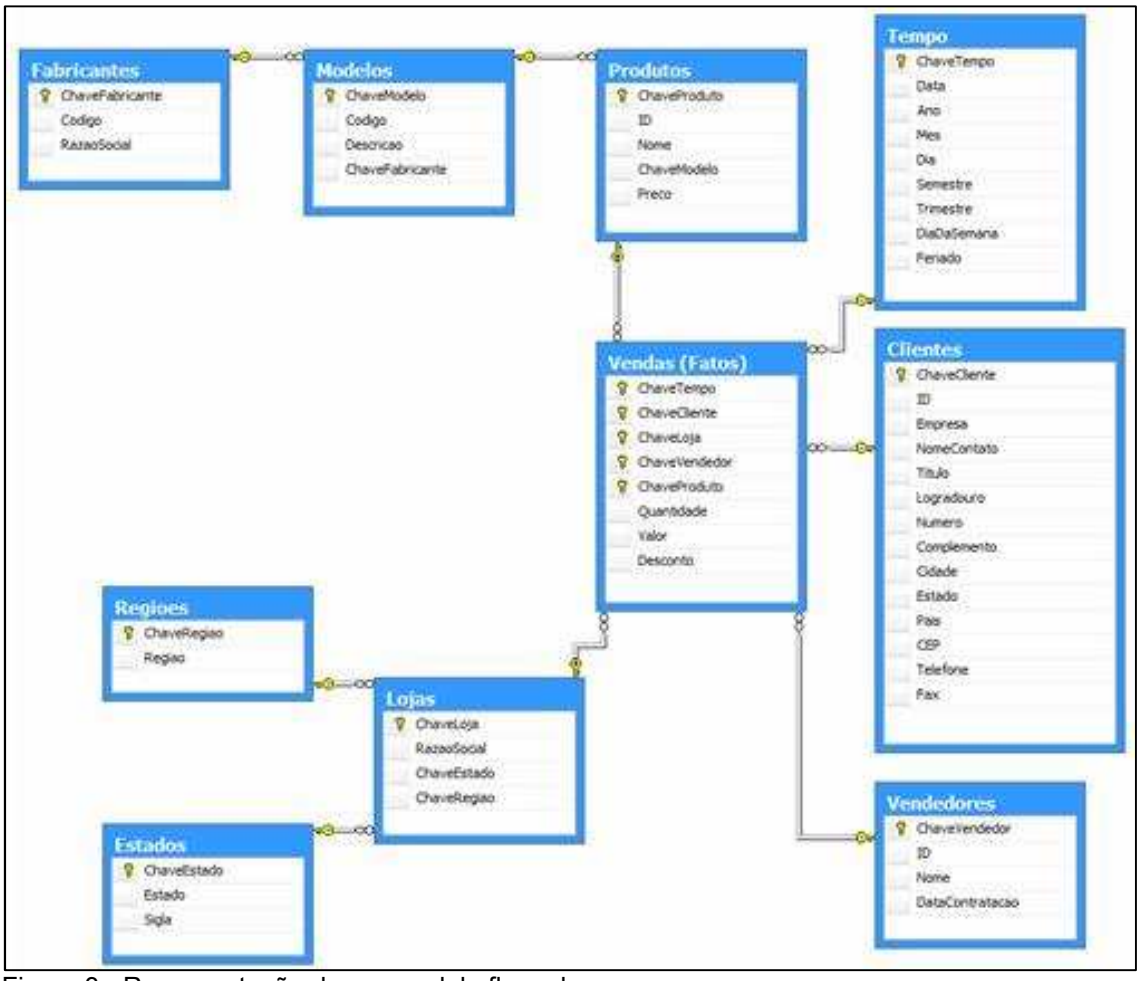


Figura 3 - Representação de um modelo floco de neve
Fonte: NARDI (2007)

A abordagem de Inmon (1997) apresenta uma estratégia de implementação conhecida como '*Top-down'*, visto que o DW é estruturado antes dos repositórios específicos, tornando a relação entre os bancos de dados mais consistentes por

partir da estrutura maior para a menor. No caso de Kimball (1998), a implementação é nomeada '*Bottom-up'* pois o DW é criado a partir dos bancos de dados específicos da organização. A vantagem nesse caso está na agilidade da criação dos repositórios menores, enquanto que a desvantagem se encontra na dificuldade de integração entre os bancos de dados para a criação do DW.

Ambas as abordagens apresentam uma relação entre o DW e fontes de informações menores. Essas fontes são denominadas *Data Mart*s.

2.2.2 *Data Mart*

Assim como *Data Warehouse*s, a definição de *Data Mart* (DM) varia de acordo com o autor. Enquanto Inmon (1997) descreve DM como sendo um sub-conjunto que alimenta um *Data Warehouse* e é adaptado às necessidades de um departamento específico (Figura 4), Kimball *et al.* (1998) alega que os DMs são a fundação de um *Data Warehouse* (Figura 5).

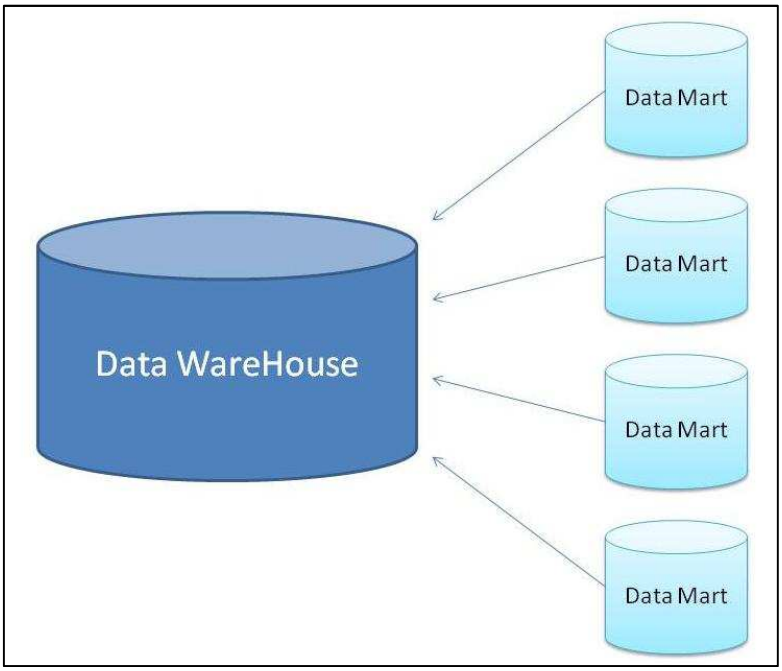


Figura 4 - *Data warehouse* e *data mart* na visão de Inmon
Fonte: O autor (2012)

Segundo Monteiro (2004), os *Data Marts* eram definidos como subconjuntos altamente agregados de dados capaz de responder uma questão de negócio específica. Tal definição levou a DMs inflexíveis e sem capacidade de integração.

Atualmente, é definido como um “conjunto flexível de dados, de preferência baseado em dados mais atômicos quanto possível e apresentados em um modelo dimensional, que é mais resistente a consultas *ad hoc* de usuários”.

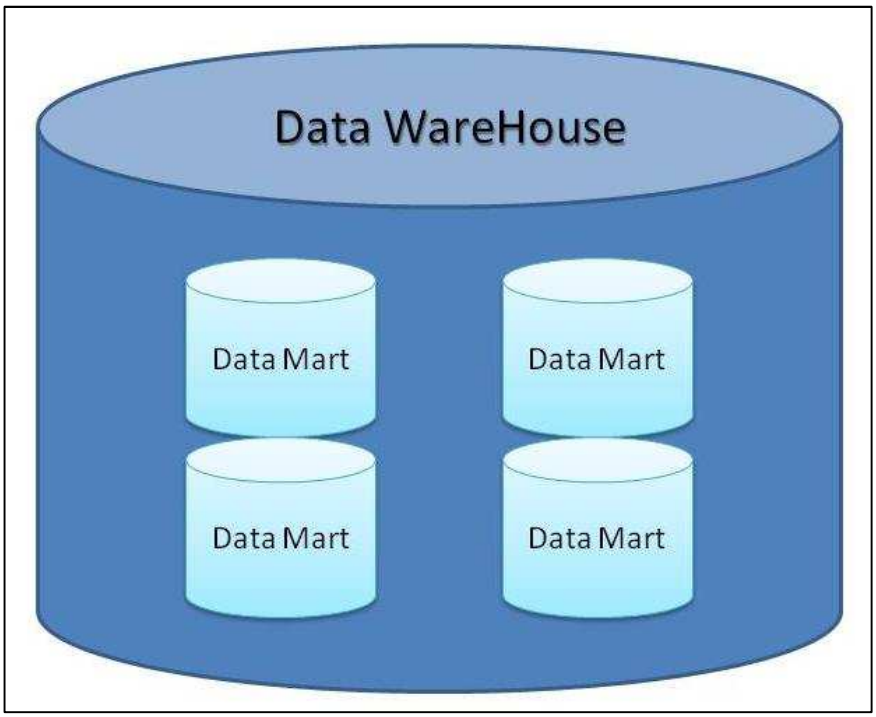


Figura 5 - *Data warehouse* e *data mart* na visão de Kimball
Fonte: O autor (2012)

2.2.3 Banco de Dados Operacionais

De acordo com Santos (2003), bancos de dados operacionais “dão suporte a todas as operações de uma organização, sendo utilizados com frequência e registrando a situação momentânea da organização”. Assim, enquanto os DWs e os DMs possuem dados estruturados de forma a fornecer informações analíticas para os níveis gerencial e estratégico da empresa, o banco de dados operacional contém os dados acerca do negócio a nível transacional, alimentando os repositórios estratégicos. Tal relacionamento é ilustrado pela Figura 6.

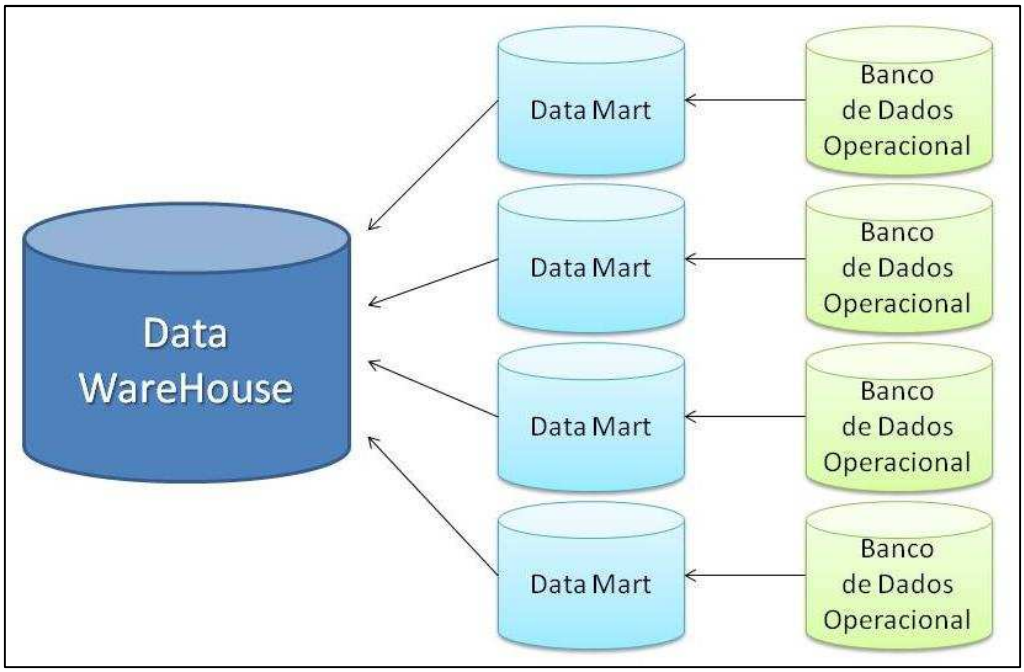


Figura 6 – Relação *data warehouse* x *data mart* x *banco de dados operacionais*
Fonte: O autor (2012).

2.3 Knowledge Discovery in Databases (KDD)

Na visão de Fayyad (1996, p.39) o *Knowledge Discovery in Databases* (KDD) ou ainda Descoberta de Conhecimento em Base de Dados (DCBD), consiste no processo não trivial de identificação de padrões compreensíveis, válidos e potencialmente úteis a partir de dados brutos. Tal processo surgiu da urgente necessidade de extrair informações do rápido e constante crescimento de dados digitais em inúmeras áreas.

Ainda segundo Fayyad (1996, p.42) o processo de KDD é iterativo e interativo e pode ser descrito em nove passos:

1º. compreensão do domínio da aplicação, do conhecimento prévio relevante e identificação do objetivo do processo de KDD;
2º. criação de um conjunto de dados alvo;
3º. limpeza dos dados e pré-processamento;
4º. redução dos dados e projeção;
5º. associação os objetivos do processo de KDD a um método específico de mineração de dados;
6º. análise exploratória e seleção de modelo e hipótese;

7º. mineração de dados;
8º. interpretação dos padrões descobertos, podendo retornar à qualquer um dos passos anteriores;
9º. ação sobre o conhecimento descoberto.

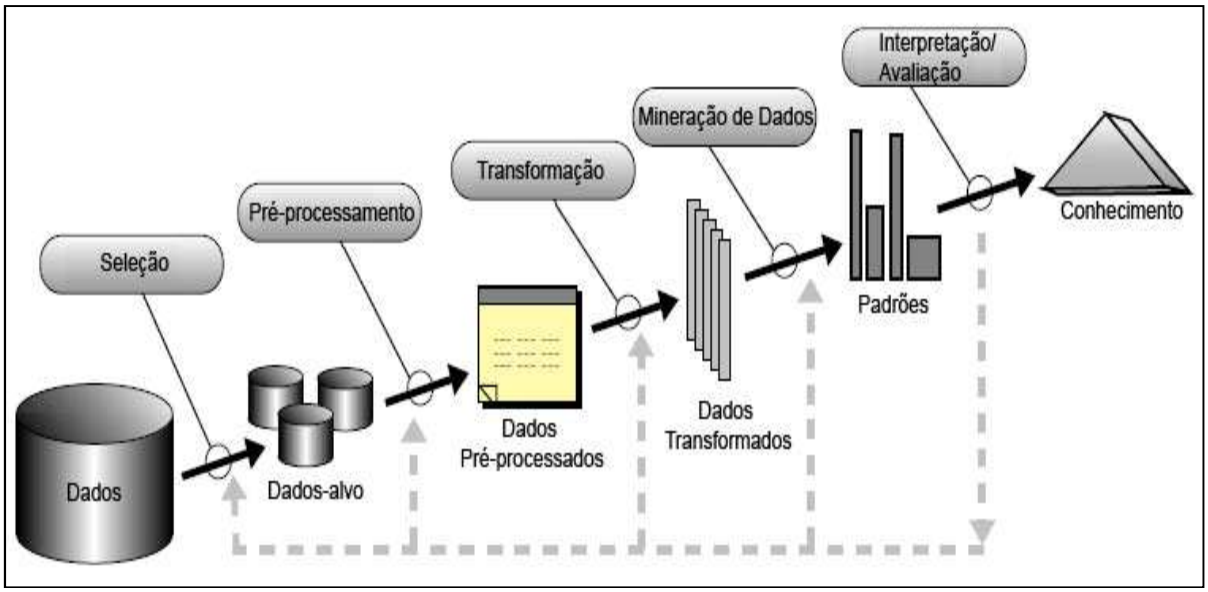


Figura 7 – Etapas do KDD
Fonte: Fayyad (1996).

As etapas podem ser vistas de maneira sintetizada na Figura 7, sendo estas:

a) seleção: etapa em que delimita-se o conjunto dos dados-alvo a serem minerados, baseando-se no tipo de conhecimento que pretende-se alcançar;
b) pré-processamento: levando em consideração que os dados podem estar disponibilizados em um formato inapropriado para o processo do KDD, como por exemplo, contendo registros duplicados ou com valores redundantes ou nulos, é feita a limpeza e adequação dos dados, eliminando inconsistências e ruídos quando necessário;
c) transformação: com os dados pré-processados, é feita a conversão dos dados para a forma mais adequada para aplicação das técnicas de mineração. O tipo de transformação depende do algoritmo que será utilizado para a extração do conhecimento;

d) mineração de dados: é considerada a principal etapa do KDD. Nesta etapa ocorre o processamento dos dados a partir de técnicas e algoritmos de mineração, buscando padrões relevantes e apresentando-os da maneira adequada para a extração do conhecimento;
e) interpretação/avaliação: após a etapa da mineração de dados é necessária a interpretação dos dados com a finalidade de verificar e validar a relevância dos padrões encontrados. Não havendo a validação, deve-se repetir as etapas anteriores do KDD até obter resultados relevantes.

A principal etapa do KDD é considerada a de Mineração de Dados, conforme explicado na seção 2.4.

2.4 Mineração de Dados

A Mineração de Dados, de acordo com Fayyad (1996) é a utilização de algoritmos específicos para a localização de padrões em grupos de dados. Além disso, permite a previsão do resultado de uma observação futura baseando-se em padrões de ocorrências anteriores. (TAN *et al.*, 2009, p.3)

Dentro do processo de MD, deve-se selecionar a tarefa, técnica e algoritmo a ser aplicado à base de dados, adaptando-os de acordo com o problema para a obtenção dos resultados desejados.

2.4.1 Aplicações

A Mineração de Dados apresenta como vantagens uma melhor capacidade de compreensão do mundo e a predição do futuro com uma maior precisão. (PIATETSKY-SHAPIRO, 2011) Pode-se também acrescentar os benefícios que o processo traz a diferentes áreas de atuação.

Dentro da área financeira, Han e Kamber (2006, p.649) citam a análise de liberação de crédito e pagamento de empréstimos para predizer as ações dos clientes que possuem características semelhantes. Outra aplicação dessa área é a descoberta de crimes financeiros, por meio de análises de fluxo financeiro, em determinado período, por algum grupo de pessoas.

Han e Kamber (2006, p.651) também citam que dentro da área de indústrias varejistas pode-se mencionar a análise da efetividade de campanhas de venda, análise de fidelidade de clientes, e recomendações de produtos, justificado por padrões entre vendas de determinados produtos. Os autores explicam que para a análise de dados biológicos, a mineração de dados pode ser aplicada, entre outros exemplos, para descoberta de padrões genéticos, ou para a predição de doenças baseadas no perfil de pacientes, possibilitando a antecipação de um tratamento.

2.4.2 Tarefas

De acordo com Amo (2004), a tarefa de mineração de dados consiste na especificação do tipo de regularidades ou categorias de padrões que se buscam nos dados. Divide-se a Mineração de Dados em duas categorias de tarefas: as preditivas e as descritivas. Tan *et al* (2006, p.8) explica que as tarefas preditivas objetivam prever o valor de um determinado atributo baseado nos valores de outros atributos, enquanto que as tarefas descritivas buscam derivar padrões que resumam os relacionamentos subjacentes nos dados.

Dessa forma, são tarefas preditivas:

a) classificação: consiste em mapear os dados de entrada em um número definido de classes, criando uma correlação entre as características dos dados e uma classe. Assim, o objetivo da tarefa é classificar dados novos e desconhecidos associando seus atributos a uma classe existente. São exemplos do uso da classificação: classificar pedidos de créditos como baixo, médio e alto risco; identificar o tratamento adequado a um paciente, baseando-se em classes de pacientes que já responderam ao tratamento;
b) regressão: também conhecida como estimativa ou predição, a tarefa de regressão é similar à classificação, porém, restringe-se a atributos numéricos. Tan *et al* (2006) define que a tarefa consiste em aprender uma função-alvo que mapeie conjuntos de atributos com um erro mínimo em saídas de valores contínuos. São exemplos de uso da regressão: estimar a renda total de uma família, previsão de índice de bolsa de valores.

São tarefas descritivas:

a) associação: caracteriza-se por identificar padrões de correlação ou co-ocorrência entre conjuntos de itens. Além do clássico exemplo da compra de produtos em um mercado, pode-se citar como uso da tarefa a associação entre sintomas apresentados por um paciente ou páginas da web acessadas juntas.
b) agrupamento: também conhecida como clusterização ou análise de clusters, consiste em identificar agrupamentos de itens de acordo com características similares ou propriedades em comum em relação a outros

grupos. Como exemplo pode-se citar o agrupamento de clientes com comportamentos de compra similares (GOEBEL e GRUENWALD, 1999).

c) sumarização: de acordo com Fayyad (1996) essa tarefa consiste em encontrar uma descrição compacta para um subconjunto de dados. Martins (2010) cita como exemplo da tarefa a identificação de características em um conjunto de perfis de usuários de uma determinada região, com faixa salarial de X reais, nível superior incompleto e que possuem residência própria.

Dentro de cada grupo de tarefas existem diversas heurísticas que serão explicadas no item 2.5.

2.5 Heurísticas

De acordo com Bogorny (2003), as heurísticas de mineração de dados atendem a diferentes propósitos e possuem vantagens e desvantagens distintas. Sua seleção depende do contexto e do domínio da aplicação, e do tipo de conhecimento que se deseja encontrar.

São algumas das principais heurísticas:

a) **algoritmos genéticos**: Pacheco (1999) descreve como algoritmos matemáticos inspirados na evolução natural e recombinação genética. O mecanismo de busca de tal técnica se baseia na teoria de Charles Darwin quanto a sobrevivência dos mais aptos;

b) **árvores de decisão**: São representações simples do conhecimento e um eficiente meio de criar classificadores que predizem classes baseadas nos valores de atributos de um conjunto de dados. (CASTANHEIRA, 2008, p.48) Netto (2007) afirma que o objetivo dos algoritmos de árvores de decisão é a criação de uma árvore na qual cada nó indica o teste de um atributo. Bogorny (2003, p. 17) explica que uma grande vantagem na técnica é que pode ser aplicada a um grande conjunto de dados e possibilita uma visão real do processo de tomada de decisão, facilitando a interpretação de seus resultados;

c) **redes neurais**: Segundo Pacheco (1999), são modelos computacionais não lineares inspirados no cérebro humano, que tentam reproduzir características como: aprendizado, associação, generalização e abstração. Castanheira (2008, p.23) complementa que as redes neurais apresentam uma estrutura paralelizada, composta por processadores simples conectados entre si, assim como no cérebro humano;
d) **baseado em regras** (Rules): Han e Kamber (2006, p.319) explicam que esse método utiliza uma série de regras Se-Então (If-Then, no original). Ou seja, são definidas regras conclusivas partindo de alguma condição, ou seja, *SE condição ENTÃO conclusão*. Bogorny (2003, p.18) aponta como uma das principais vantagens da heurística a facilidade de interpretação dos resultados, facilidade de incorporação de conhecimento explicito nas regras e facilidade de armazenamento das regras;
e) **classificadores bayeseanos**: De acordo com Han e Kamber (2006, p.310) são classificadores estatísticos, baseado no teorema de Bayes, que podem predizer resultados e classificações utilizando probabilidade. Estudos apontaram que os classificadores Bayeseanos podem ser comparados em performance com árvores de decisão e redes neurais, apresentando alta velocidade e acurácia quando aplicado em grandes bases de dados;
f) **lógica nebulosa** (*Fuzzy*): Inspirado no processamento lingüístico, a Lógica Nebulosa, ou *Fuzzy*, tem por objetivo reproduzir o raciocínio humano e desenvolver sistemas para tomadas de decisões racionais em meio a informações incompletas, imprecisas ou não totalmente confiáveis. (PACHECO, 1999).

No âmbito de cada heurística apresentada existem diversos métodos, dentre os quais foram escolhidos os métodos C4.5, visto que é um algoritmo básico da tarefa de classificação, e o método Apriori, um dos algoritmos mais utilizados para descoberta de regras de associação.

2.5.1 C4.5

O algoritmo C4.5 é uma extensão do algoritmo *Iterative Dichotomiser 3* (ID3), ambos desenvolvidos por Ross Quinlan, e que tem por função construir uma árvore de decisão partindo de uma raiz calculada. O C4.5 foi desenvolvido para suprir as deficiências apresentadas pelo ID3. Entre as melhorias do algoritmo, pode-se citar a manipulação de valores ausentes, a remoção de 'galhos' redundantes da árvore de decisão e o manuseio de valores numéricos, categorizando-os quando esses forem contínuos (KOHAVI e QUINLAN, 1999). Este processo de categorização dos atributos é também chamado de discretização, e tem por função transformar os dados para atender os requisitos de determinados algoritmos (TAN *et al.*, 2009).

A construção da árvore de decisão se inicia pelo cálculo da entropia e do ganho de informação dos atributos do conjunto de dados a serem minerados. Assim, define-se primeiro a entropia através da seguinte fórmula:

$$H(S) = -\sum_{i=1}^{c} p_i \log_2 p_i \quad (1)$$

sendo:

S = conjunto de classes;

c = número de classes;

p_i = freqüência da classe i;

Com a entropia calculada, deve-se descobrir o ganho de informação, definido da seguinte maneira:

$$Ganho\ (S, A) = H(S) - \sum_{i=1}^{c} p\left(t_i^A\right) H(S|t_i^A) \quad (2)$$

onde:

H(S) = entropia calculada;

c = número de classes;

i = i-ésimo valor do atributo A;

p$\left(t_i^A\right)$ = frequência do i-ésimo valor do atributo A;

$H(S|t_i^A)$ = entropia do i-ésimo valor.

A raiz da árvore é definida pelo classificador com o maior ganho de informação calculado. Para o segundo nível deverá ser feito os cálculos de entropia e ganho novamente para cada uma das opções do primeiro classificador. Assim, no segundo nível ficará o atributo com o maior ganho e assim sucessivamente para os outros níveis da tabela.

Um exemplo de aplicação do C4.5 poderia ser desenhar a árvore de decisão para o Quadro 1.

Quadro 1 - Exemplo de aplicação do algoritmo C4.5

Faixa Etária	Sexo	Tamanho	Compra Camiseta?
A	F	G	Sim
A	M	M	Não
B	M	M	Não
B	F	P	Não
A	M	G	Sim

Fonte: O autor (2011) , baseado em TSUNODA (2010)

a) Cálculo de Entropia do conjunto a se obter uma resposta (Meta):

$$H(S) = -\sum_{i=1}^{2} p_i \log_2 p_i$$

$$H(S) = -\frac{2}{5}\log_2 \frac{2}{5} - \frac{3}{5}\log_2 \frac{3}{5}$$

$$H(S) = -0{,}4 \log_2 0{,}4 - 0{,}6 \log_2 0{,}6$$

$$H(S) = -0{,}4(-1{,}3220) - 0{,}6(-0{,}7370)$$

$$H(S) = 0{,}5288 + 0{,}4422$$

$$H(S) = \mathbf{0{,}9710}$$

Em seguida, devem-se definir os ganhos de informação para os atributos ‘Faixa Etária’, ‘Sexo’ e ‘Tamanho’:

b1) Faixa Etária:

$$H(A, Sim) = -\frac{2}{3}\log_2 \frac{2}{3} = 0{,}3900$$

$$H(B, Sim) = 0$$

$$H(A, Não) = -\frac{1}{3}\log_2\frac{1}{3} = 0{,}5284$$

$$H(B, Não) = -\frac{2}{2}\log_2\frac{2}{2} = 0$$

$$Ganho(FaixaEtária) = 0{,}9710 - \tfrac{3}{5}\,(0{,}3900 + 0{,}5284) - \tfrac{2}{5}(0+0) = \mathbf{0{,}4200}$$

b2) Sexo:

$$H(F, Sim) = -\frac{1}{2}\log_2\frac{1}{2} = 0{,}5000$$

$$H(M, Sim) = -\frac{1}{3}\log_2\frac{1}{3} = 0{,}5284$$

$$H(F, Não) = -\frac{1}{2}\log_2\frac{1}{2} = 0{,}5000$$

$$H(M, Não) = -\frac{2}{3}\log_2\frac{2}{3} = 0{,}3900$$

$$Ganho(Sexo) = 0{,}9710 - \tfrac{2}{5}\,(0{,}5 + 0{,}5) - \tfrac{3}{5}(0{,}5284 + 0{,}3900) = \mathbf{0{,}02}$$

b3) Tamanho:

$$H(P, Sim) = -\frac{0}{1}\log_2\frac{0}{1} = 0$$

$$H(M, Sim) = -\frac{0}{2}\log_2\frac{0}{2} = 0$$

$$H(G, Sim) = -\frac{2}{2}\log_2\frac{2}{2} = 0$$

$$H(P, Não) = -\frac{1}{1}\log_2\frac{1}{1} = 0$$

$$H(M, Não) = -\frac{2}{2}\log_2\frac{2}{2} = 0$$

$$H(G, Não) = -\frac{0}{2}\log_2\frac{0}{2} = 0$$

$$Ganho(Tamanho) = 0{,}9710 - \tfrac{1}{5}\,(0+0) - \tfrac{2}{5}(0+0) - \tfrac{2}{5}(0+0) = \mathbf{0{,}9710}$$

Logo, o maior ganho é o do atributo Tamanho, se tornando o primeiro classificador da árvore.

Para definir o segundo atributo classificador, é necessário desmontar a tabela conforme as possibilidades da raiz, calculando uma nova entropia e novos ganhos, ou seja:

a) Para o tamanho P:

Quadro 2 - Registros contendo tamanho P

Faixa Etária	Sexo	Tamanho	Compra Camiseta?
B	F	P	Não

Fonte: O autor (2011), baseado em TSUNODA (2010)

$$H(S) = \mathbf{0}$$

b) Para o tamanho M:

Quadro 3 - Registros contendo tamanho M

Faixa Etária	Sexo	Tamanho	Compra Camiseta?
A	M	M	Não
B	M	M	Não

Fonte: O autor (2011) , baseado em TSUNODA (2010)

$$H(S) = \mathbf{0}$$

c) Para o tamanho G:

Quadro 4 - Registros contendo tamanho G

Faixa Etária	Sexo	Tamanho	Compra Camiseta?
A	F	G	Sim
A	M	G	Sim

Fonte: O autor (2011) , baseado em TSUNODA (2010)

$$H(S) = \mathbf{0}$$

Neste exemplo os atributos ‘Faixa Etária’ e ‘Sexo’ não representaram ganho de informação a partir do segundo nível, visto que somente o tamanho é o suficiente

para definir se haverá a compra ou não. Assim, a árvore de decisão ficará como mostra a Figura 8.

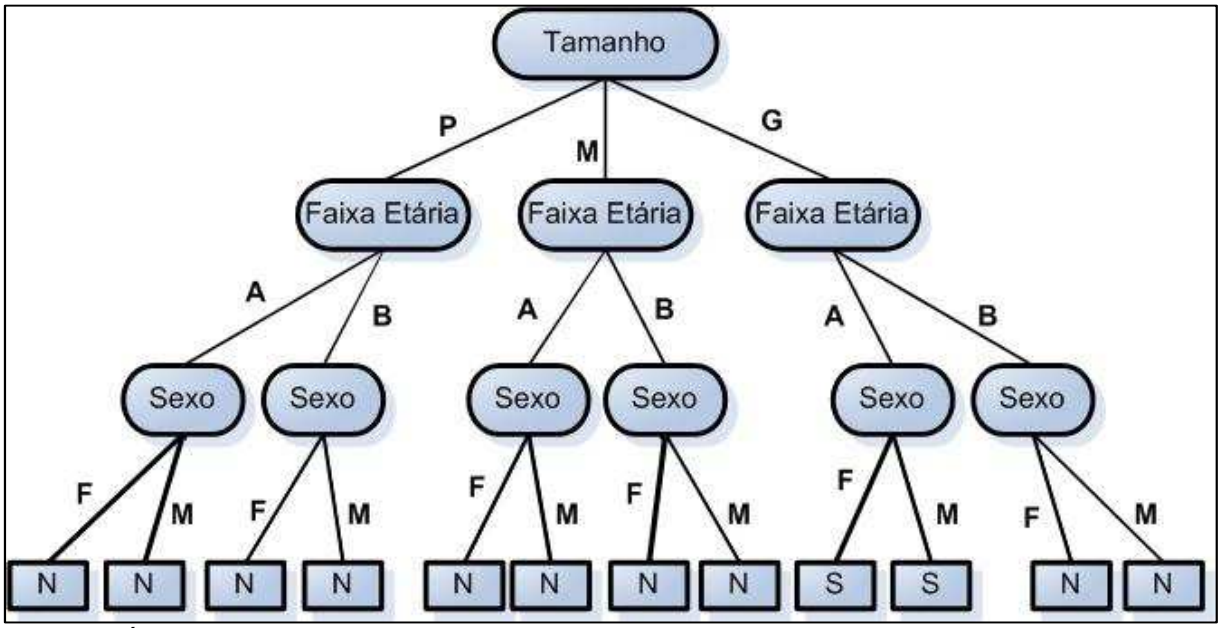


Figura 8 - Árvore de decisão gerada pelo algoritmo ID3
Fonte: O autor (2011) , baseado em TSUNODA (2010)

Porém, o C4.5 já apresenta como resultado a árvore podada, reduzindo todos os dados redundantes, como visto na Figura 9.

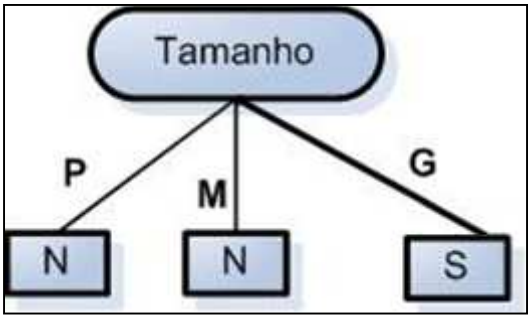


Figura 9 - Árvore de decisão gerada pelo algoritmo C4.5 com poda
Fonte: O autor (2011) , baseado em TSUNODA (2010)

Assim, obtém-se como resposta que caso o tamanho da camiseta seja P ou M não haverá a compra da mesma. Sendo do tamanho G, haverá a compra.

2.5.2 Apriori

De acordo com Rakesh Agrawal (1994), o Apriori é um algoritmo de associação que busca por padrões de frequência em grupos de itens. Para isso, deve-se descobrir o suporte e a confiança dos itens e combinações de itens.

Para estabelecer uma regra de associação, é necessário primeiro descobrir o suporte dos itens individualmente, da seguinte forma:

$$FSup = \frac{|X\ U\ Y|}{N} \tag{3}$$

sendo:

X = número de ocorrências do item 1;

Y = número de ocorrências do item 2;

N = número do total de registros.

Caso o suporte dos itens atenda ao suporte mínimo estabelecido, deve-se combiná-los 2 x 2 e calcular o suporte para cada uma das combinações. Havendo alguma combinação que ultrapasse o suporte mínimo, devem ser feitas novas combinações, desta vez 3 x 3, com os itens que possuem, individualmente, o suporte mínimo e calcular o suporte dos novos grupos de itens. Assim sucessivamente até que não retorne nenhum grupo com o suporte mínimo. Então, deve-se calcular a confiança dos grupos, através da seguinte fórmula:

$$FConf = \frac{|X\ U\ Y|}{X} \tag{4}$$

onde:

X = número de ocorrências do item 1;

Y = número de ocorrências do item 2;

As combinações que atingirem tanto o valor de suporte mínimo quanto o de confiança mínima estabelecerão as regras de associação, resposta do problema.

Para ilustrar o algoritmo, tem-se o seguinte exemplo:

Um determinado mercado deseja descobrir quais são as relações entre os produtos: Batata, Tomate, Cenoura e Banana, baseado nas seguintes compras:

1. Batata, Tomate;
2. Tomate, Cenoura, Batata;
3. Banana, Batata, Cenoura;
4. Cenoura, Banana;
5. Não comprou nenhum dos itens pesquisados;
6. Batata, Cenoura, Banana.

Suporte Mínimo = 40%

Confiança Mínima = 70%

Assim, calcula-se o suporte dos produtos individualmente:

Quadro 5 - Suporte dos itens individualmente

	Batata	Tomate	Cenoura	Banana
Compra 1	X	X		
Compra 2	X	X	X	
Compra 3	X		X	X
Compra 4			X	X
Compra 5				
Compra 6	X		X	X
SUPORTE:	**4/6**	**2/6**	**4/6**	**3/6**

Fonte: O autor (2011) , baseado em TSUNODA (2010)

Os itens que atenderam o suporte mínimo foram: Batata, Cenoura e Banana, portanto, é feita uma combinação 2 x 2 efetuando uma nova contagem das ocorrências.

Quadro 6 - Suporte dos itens em combinação 2 X 2

Combinação	**Suporte**
Batata, Cenoura	3/6
Batata, Banana	2/6
Cenoura, Banana	3/6

Fonte: O autor (2011) , baseado em TSUNODA (2010)

Como houve pelo menos uma combinação que atingiu o suporte mínimo, deve-se repetir o processo combinando os itens 3 x 3:,

Quadro 7 - Suporte dos itens em combinação 3 X 3

Combinação	Suporte
Batata, Cenoura, Banana	2/6

Fonte: O autor (2011) , baseado em TSUNODA (2010)

Não havendo combinações com o suporte mínimo, deve-se retornar ao último grupo de itens e calcular a confiança somente das combinações que atingiram o suporte mínimo:

Quadro 8 - Confiança das combinações possíveis

Combinação	Suporte	Confiança
Batata → Cenoura	3/6	3/4
Cenoura → Banana	3/6	3/4
Cenoura → Batata	3/6	3/4
Banana → Cenoura	3/6	3/3

Fonte: O autor (2011) , baseado em TSUNODA (2010)

Como todas as combinações possíveis atingiram o suporte e a confiança mínima, então a resposta para o exemplo é:

- Se Batata então Cenoura;
- Se Cenoura então Banana;
- Se Cenoura então Batata;
- Se Banana então Cenoura.

Assim, a partir da conceituação e exemplos, são descritas as linguagens utilizadas para a aplicação dos algoritmos no sistema desenvolvido.

2.6 Linguagem de Programação Web

2.6.1 *HyperText Markup Language* (HTML)

Desenvolvido por Tim Berners-Lee, o *Hypertext Markup Language* (HTML) é uma linguagem para formatação e publicação de conteúdo na web que baseia-se no conceito de hipertexto, ou seja, na ligação dos conteúdos através de links tornando a navegação não linear. (http://www.w3c.br)

A estrutura da linguagem HTML consiste do uso de etiquetas (*tags*) que delimitam e definem alguma característica a uma porção do conteúdo da página. As *tags* principais de uma página HTML são:

- <html>: indica que todo o conteúdo dentro da etiqueta deverá ser lido como um código HTML;

- <head>: é o cabeçalho do documento, apresenta informações que serão lidas pelo navegador, como o texto a ser exibido no título da janela ou a codificação dos caracteres a serem lidos. É nessa etiqueta que são chamados scripts que não estão escritos no documento;

- <body>: são as informações imprimidas na tela. Toda a informação que será exibida no navegador estará obrigatoriamente dentro da *tag* <body>.

Dentro da *tag* <body>, diversas outras etiquetas são utilizadas para definir como o conteúdo será apresentado. Assim sendo, a *tag* <p> delimita parágrafos, a *tag* <table> cria uma tabela, a *tag* <form> cria formulários, e assim por diante.

Para cada etiqueta, o HTML permite a inserção de atributos que fornecem mais informações sobre o objeto delimitado. Dentre as possibilidades, pode-se descrever um nome, uma classe com características próprias, um identificador único, e até mesmo uma formatação detalhada do conteúdo.

2.6.2 *PHP*: *Hypertext Preprocessor*

De acordo com a página oficial da linguagem (http://www.php.net/), o PHP, acrônimo que significa *PHP*: *Hypertext Preprocessor*, é uma linguagem embutida em um código HTML, de modo a gerar páginas dinâmicas. Foi desenvolvido por Andi Gutmans e Zeev Suraski em 1997, baseado em uma ferramenta anterior chamada

Personal Home Page/ *Forms Interpreter* (PHP/FI),e melhorado, acrescentando funções e extensões para comunicação com outras ferramentas.

Winckler e Pimenta (2002) explicam que uma arquitetura cliente/servidor é uma rede de computadores em que de um lado, chamado de cliente, está o navegador e que faz solicitações de informações para o outro lado, chamado de servidor.

Por se tratar de uma linguagem que interage com o servidor, o PHP permite diversas funcionalidades, dentre elas:

- desenvolver aplicativos web;
- efetuar transações com bancos de dados;
- trocar dados com um servidor web, como recebimento e envio de *cookies*;
- gerar arquivos em vários formatos;
- criar e manipular conteúdos dinamicamente;

Descreve-se também na página oficial que é possível com o PHP desenvolver programação tanto estruturada quanto orientada a objetos, ao mesmo tempo em que interage com outras linguagens. A estrutura da linguagem permite que o PHP seja inserido em qualquer trecho do código. A partir do momento que o servidor reconhece a *tag* “<?php” ele inicia a execução do código, parando ao deparar-se com a *tag* de fechamento “?>”.

3 PROCEDIMENTOS METODOLÓGICOS

A pesquisa se caracteriza como exploratória definida por Lakatos e Marconi (2001, p.188) como investigações empíricas com tripla finalidade: desenvolver hipóteses, aumentar a familiaridade do cientista com um ambiente, fato ou fenômeno, e modificar e clarificar conceitos. Gil (2002, p.45) complementa que o objetivo da pesquisa exploratória é o aprimoramento de ideias ou a descoberta de intuições, como formas para auxiliar a elaboração de hipóteses.

O método a ser utilizado na pesquisa é o estudo de caso, entendido como uma metodologia ou como a escolha de um objeto de estudo definido pelo interesse em casos individuais. Visa à investigação de um caso específico, bem delimitado, contextualizado em tempo e lugar para que se possa realizar uma busca circunstanciada de informações. (VENTURA, 2007, p.384).

De acordo com Gil (2002, p.58) a maior utilidade dos estudos de caso é verificada nas pesquisas exploratórias, visto sua flexibilidade para construção de hipóteses ou reformulação de problemas.

Assim sendo, esse projeto será executado em cinco etapas, conforme as especificações a seguir:

a) definição dos conceitos básicos que permeiam a pesquisa;
    - revisão de literatura pertinente à área de Mineração de Dados:
b) análise dos softwares existentes:
    - comparação entre as características dos softwares;
c) estabelecimento das características da ferramenta:
    - definição de especificações do sistema;
    - definição da linguagem utilizada;
d) implantação do sistema em um servidor web:
    - estabelecimento de um ambiente para implementação da ferramenta;
    - implementação dos algoritmos;
    - definição de aspectos visuais.
e) análise dos resultados:
    - validação dos resultados;
    - comparação com resultados das outras ferramentas.

3.1 Ferramentas

Existem, atualmente, várias ferramentas para Mineração de Dados. Baseado em Goldschmidt e Passos (2005, p.120) elaborou-se a tabela abaixo, listando algumas ferramentas *freeware/shareware* de código aberto, mais utilizadas atualmente em ambientes acadêmicos, disponíveis para download no site KDNuggets. As ferramentas analisadas utilizam como sistema operacional padrão o *Microsoft Windows*®.

Os softwares a serem analisados são:

a) Weka;
b) *RapidMiner*;
c) TANAGRA;
d) Linguagem R;
e) Orange.

Os critérios de avaliação foram definidos com base em Goldschmidt e Passos (2005, p.120) e no questionário para avaliação de software elaborado pela Universidade Federal do Rio de Janeiro, levando em consideração a funcionalidade, confiabilidade, usabilidade e eficiência dos softwares. Assim, ficaram definidos os seguintes critérios:

a) tamanho do arquivo de instalação em *quilobytes (KB)*;
b) linguagem de desenvolvimento;
c) funcionamento em ambiente web;
d) possibilidade de uso do algoritmo C4.5;
e) possibilidade de uso do algoritmo Apriori;
f) suporte a diversos formatos de arquivos de entrada;
g) interface em língua portuguesa;
h) apresentação do passo-a-passo do processamento;
i) inclusão de novos métodos/ operações;
j) última atualização no ano corrente.

Relacionando os critérios às ferramentas, tem-se o Quadro 9.

Quadro 9 - Comparativo entre softwares de mineração de dados

Critério de Avaliação \ Ferramenta Avaliada	Weka	RapidMiner	TANAGRA	Linguagem R	Orange
Tamanho do arquivo de instalação em *quilobytes*	21.373	53.088	2.725	39.271	69.700
Linguagem de desenvolvimento	Java	Java	Delphi 6	Linguagem R	Python
Funciona em ambiente web	Não	Não	Não	Não	Não
Possui aplicação do algoritmo C4.5	Sim	Não	Sim	Sim	Não
Possui aplicação do algoritmo Apriori	Sim	Sim	Sim	Sim	Sim
Suporta diversos formatos de arquivos de entrada	Sim (*.arff,*.arff.gz, *.names,*.data *.csv,*.libsvm, *.dat,*.bsi,*.xrff *.xrff.gz)	Sim (*.cvs,*.xls, *.adp,*.dbf)	Sim (*.tdm,*.bdm, *.txt,*.xls,*.arff)	Sim (*.R,*.q, formatos de texto, formatos lidos pela função “read.”)	Sim (*.tab, *.txt, *.data, *.dat, *.rda, *.rdo, *.arff, *.xml, *.svm, *.basket)
Interface em língua portuguesa	Não	Não	Não	Sim	Não
Apresentação do passo-a-passo do processamento	Não	Não	Não	Não	Não
Inclusão de novos métodos/ operações	Sim	Depende de plugins disponíveis	Não	Depende da instalação das bibliotecas	Depende da instalação de *widgets*
Última atualização no ano corrente	Sim	Não	Sim	Sim	Sim

Fonte: O autor (2012)

A partir do Quadro 9 percebe-se que não há intenção por parte dos desenvolvedores em apresentar o processamento detalhado para o usuário, somente os resultados do algoritmo aplicado. As ferramentas são desenvolvidas dessa forma para que não haja um uso elevado do processador que comprometa a agilidade na apresentação dos resultados.

Da mesma forma, pode-se observar que nenhuma das ferramentas funciona em ambiente web tendo como finalidade evitar o risco de sobrecarga do sistema, dependendo do tamanho da base de dados a ser minerada, além dos custos para se manter uma infraestrutura web.

Todas as ferramentas apresentam o Apriori como opção de aplicação, evidenciando a alta utilização do algoritmo, representante principal das regras de associação.

É possível também perceber a falta de padronização para arquivos de bancos de dados, visto que não há um formato de entrada comum a todos os aplicativos.

Por fim, avalia-se que a evolução desses softwares ocorre à medida que aumenta o número de usuários colaborativos, devido à característica de código aberto, presente em todas as ferramentas citadas.

3.2 Limitações de pesquisa

A pesquisa se limita ao desenvolvimento de um sistema e comparação de seus resultados com softwares que possuem como algoritmos disponíveis o C4.5 e/ou o Apriori, sob a característica de código aberto e instalação em sistemas operacionais Windows®.

Em relação à ferramenta a ser desenvolvida, delimita-se a aplicação dos algoritmos C4.5 e Apriori para uso didático, suportando conferência de exercícios com tabelas que não contenham muitos registros.

4 EXPERIMENTO

Nesta seção serão descritas as especificações do sistema baseadas na análise comparativa do quadro 9, a utilização do sistema, funcionamento e análise dos dados resultantes da ferramenta desenvolvida.

4.1 Especificação do sistema

Definem-se as seguintes especificações do sistema, de acordo com os critérios estabelecidos para avaliação dos softwares:

a) linguagem PHP e HTML para desenvolvimento e apresentação do sistema;
b) funcionamento apenas em ambiente web;
c) possibilidade de uso apenas dos algoritmos C4.5 e Apriori;
d) suporte aos formatos *.txt e *.csv para padrão de arquivos de entrada, conforme formatação vista na Figura 10 por apresentarem um formato visualmente organizado e de fácil edição,;
e) interface em língua portuguesa;
f) apresentação do passo-a-passo do processamento.

```
Aparencia,Temperatura,Umidade,Vento,Jogo
Sol,23,97,Falso,Não
Sol,13,96,Verdade,Não
Encoberto,17,98,Falso,Sim
Chuvoso,9,99,Falso,Sim
Chuvoso,20,90,Falso,Sim
Chuvoso,12,91,Verdade,Não
Encoberto,8,90,Verdade,Sim
Sol,18,98,Falso,Não
Sol,15,92,Falso,Sim
Chuvoso,22,93,Falso,Sim
Sol,21,91,Verdade,Sim
Encoberto,20,97,Verdade,Sim
Encoberto,25,90,Falso,Sim
Chuvoso,19,100,Verdade,Não
```

Figura 10 – Formatação do arquivo de entrada
Fonte: O autor (2012)

Os demais critérios não tiveram necessidade de especificação.

Descreve-se na Imagem 11 todo o sistema, diagramado em formato de fluxograma com a finalidade de demonstrar a ordem lógica do funcionamento da ferramenta.

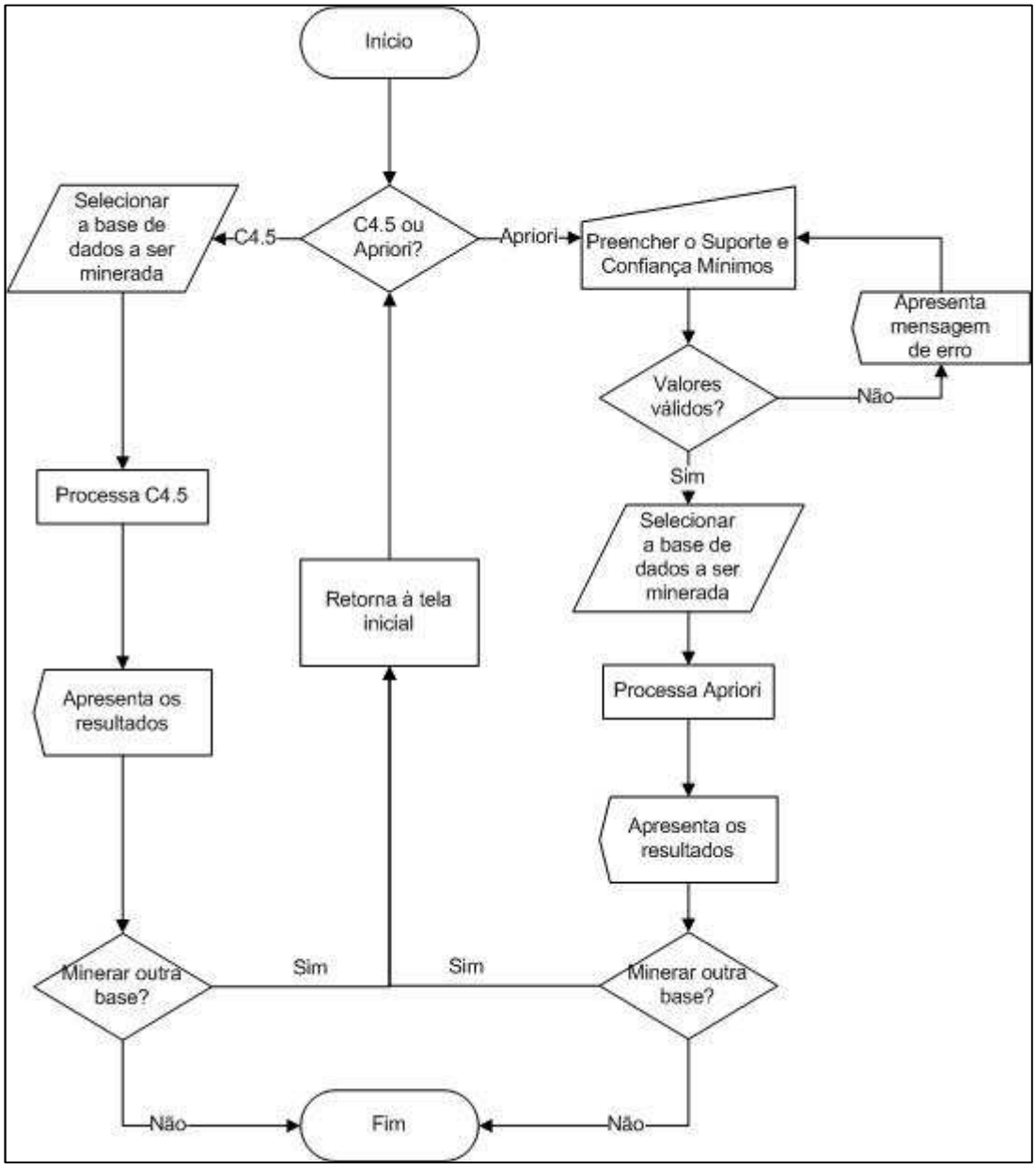


Figura 11 – Diagrama de funcionamento do sistema
Fonte: O autor (2012)

4.2 Utilização

Ao acessar a página principal, o usuário deve selecionar qual algoritmo ele deseja aplicar, como pode ser visto na Figura 12.

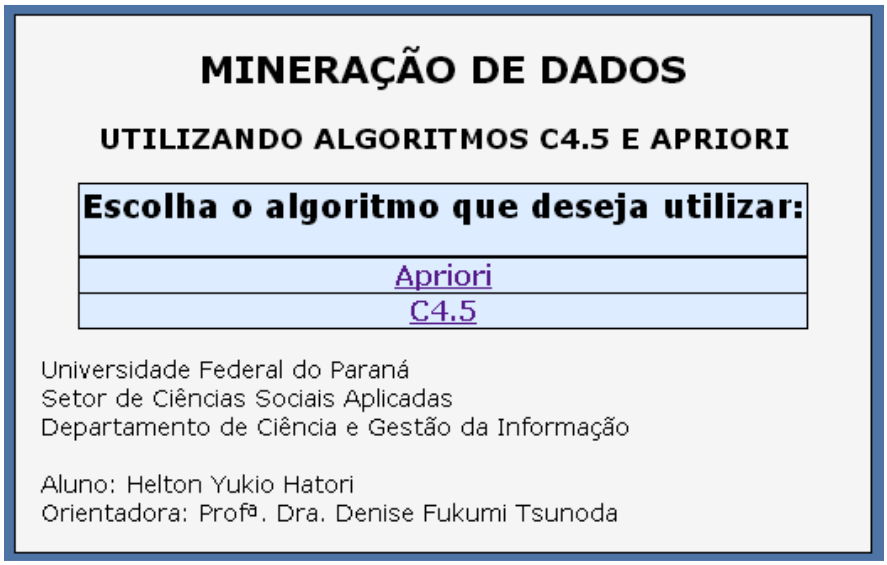


Figura 12 – Tela inicial do sistema
Fonte: O autor (2012)

Ao clicar no algoritmo, é redirecionado para a página específica do algoritmo. Estão descritas nessa página as principais informações sobre o algoritmo, uma explicação sobre os parâmetros de entrada dos dados e o formulário para inserção dos dados necessários para a execução do algoritmo (Figura 13). Como a página Apriori precisa da inserção de valores numéricos em percentual, criou-se uma validação nos campos, permitindo somente valores entre 0 e 100. Sendo diferente do estipulado, o sistema exibe uma janela *pop-up* informando o formato de entrada (Figura 14).

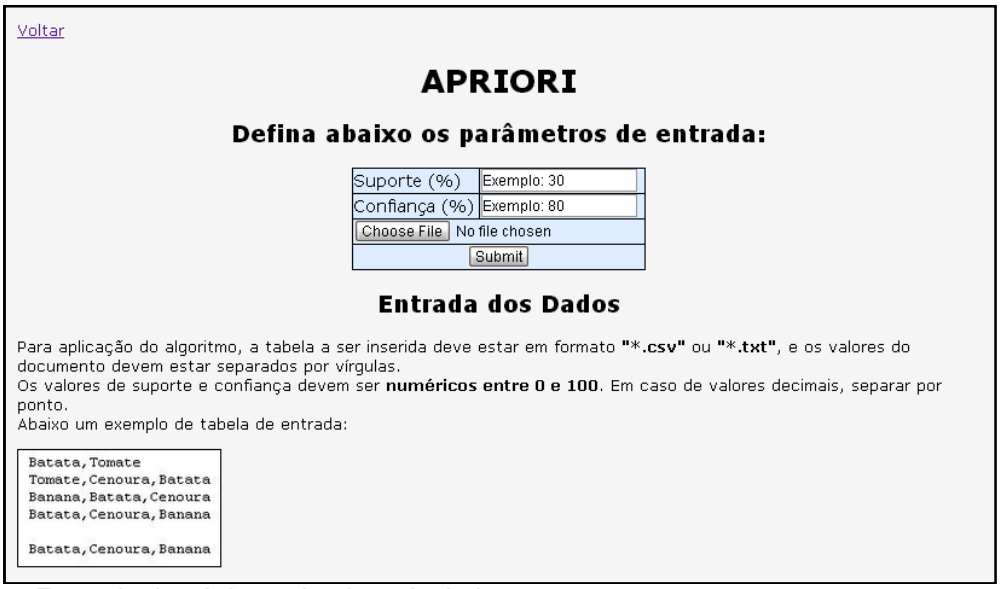


Figura 13 – Exemplo de página - algoritmo Apriori
Fonte: O autor (2012)

Para uma melhor navegação, foram inseridos em todas as páginas do sistema links para retornar à página anterior, posicionada sempre no canto superior esquerdo para facilitar a visualização do usuário.

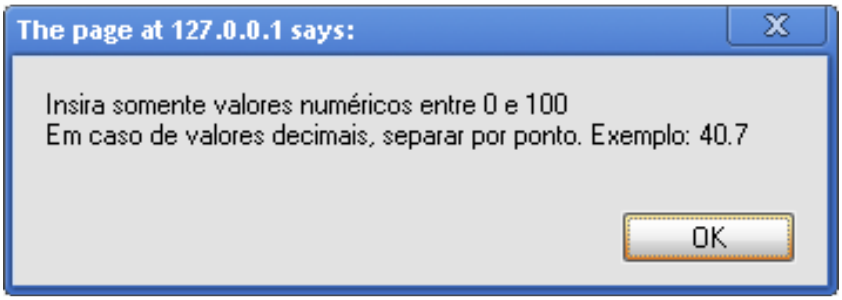


Figura 14 – Janela pop-up de validação dos campos suporte e confiança
Fonte: O autor (2012)

Ao inserir os dados corretamente no formulário e dar continuidade no processo, o sistema automaticamente executa o algoritmo solicitado, apresentando os resultados assim que processados.

4.3 Funcionamento, Apresentação e Análise dos Resultados

Para fins de validação, foram utilizadas as mesmas tabelas descritas nos seções 2.5.1 e 2.5.2 anteriormente citados.

Para o algoritmo Apriori, apresenta-se os dados da tabela escolhida, juntamente com a contagem das linhas da tabela e as informações de Suporte e Confiança Mínimos inseridas pelo usuário. Na sequência são exibidos os cálculos de Suporte para cada um dos itens até o momento que as combinações não apresentem um Suporte superior ao mínimo estabelecido. Após isso são apresentadas as regras que possuem Confiança maior ou igual à Confiança Mínima inserida.

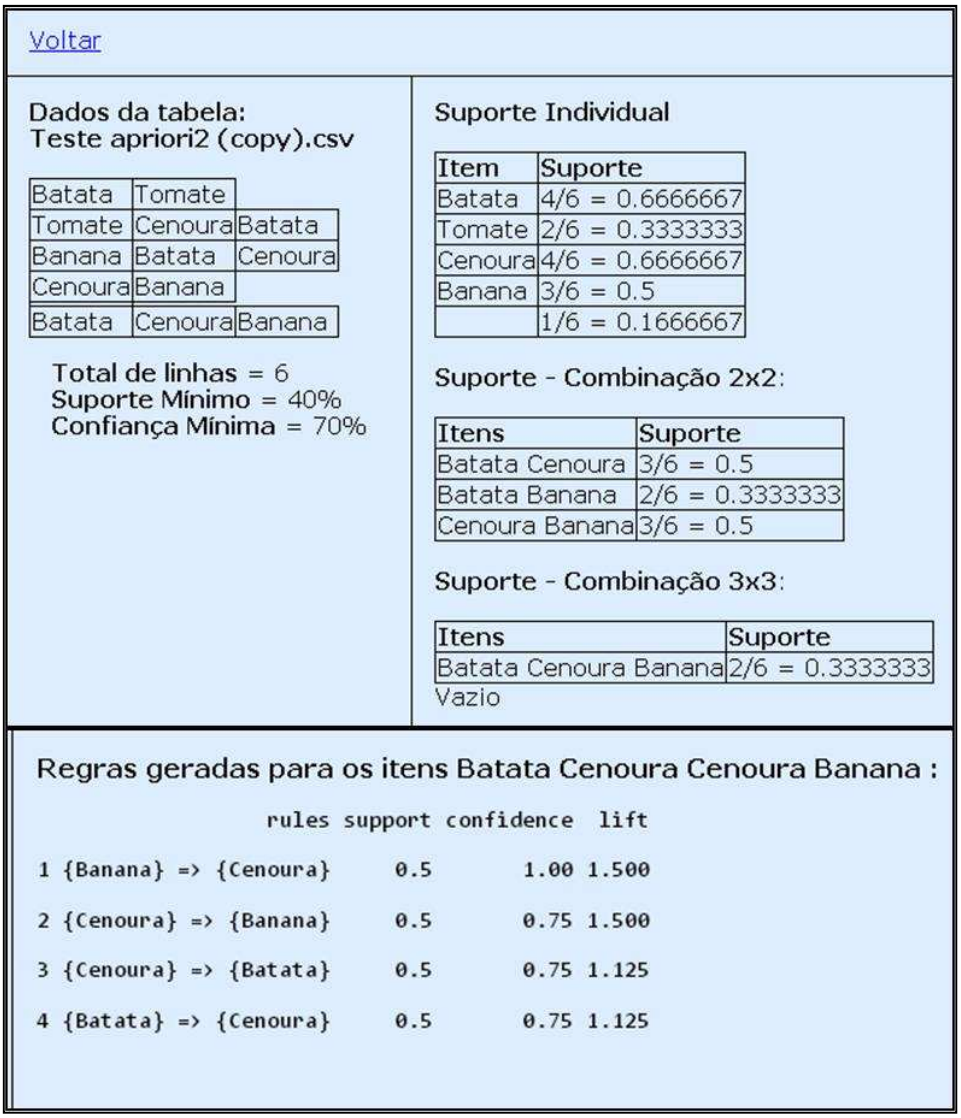


Figura 15 – Exibição dos dados, cálculos e resultados do algoritmo Apriori
Fonte: O autor (2012)

No caso do algoritmo C4.5, são apresentados os dados da tabela, visto que somente a base é inserida pelo usuário no sistema. Na coluna “Processo”, são feitos e exibidos os cálculos de Entropia e Ganho de Informação dos atributos, considerando sempre a última coluna da tabela como sendo o Atributo Meta. Após os cálculos iniciais, define-se qual é a raiz da árvore, e partindo dela é remontada a tabela para cada um de seus valores, conforme mostra a Figura 16.

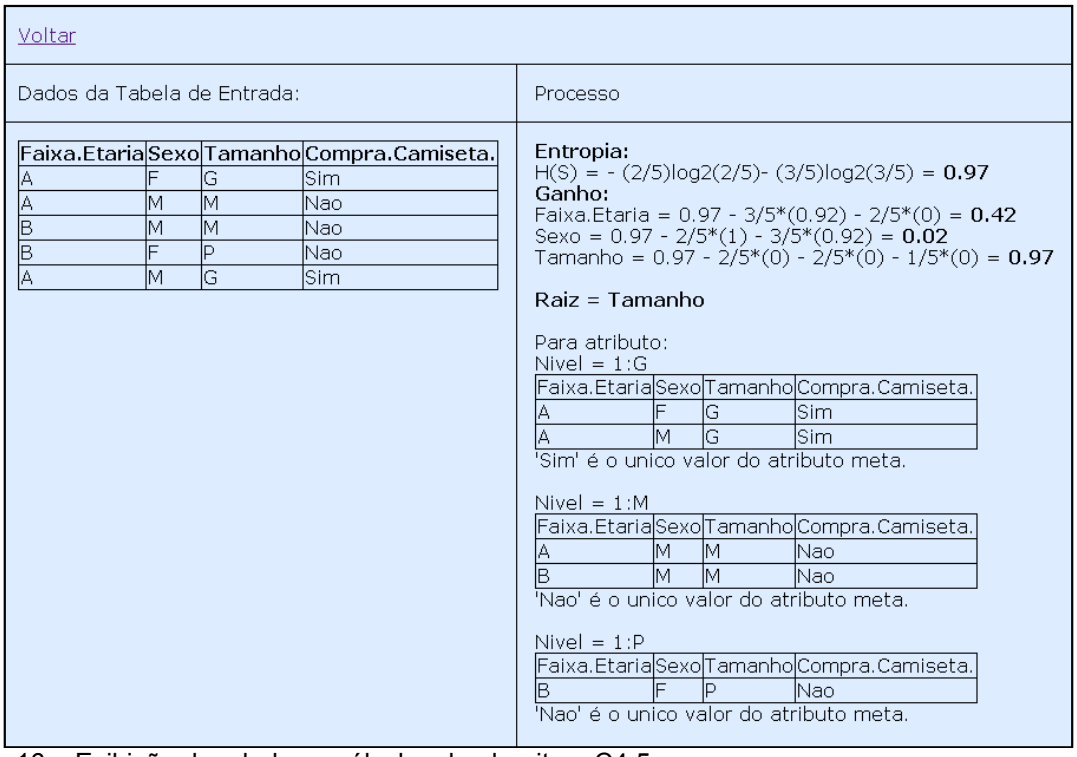


Figura 16 – Exibição dos dados e cálculos do algoritmo C4.5
Fonte: O autor (2012)

Todo o processo de cálculos é recursivo, então a tabela pode ser remontada inúmeras vezes até que se chegue a uma tabela que esta não possa / precise ser subdivida.

A apresentação da Árvore de Decisão ocorre possivelmente de duas formas: em forma de tabela aninhada com valores textuais, e em forma gráfica (Figura 17). Em ambos os casos a árvore é apresentada podada, pois o cálculo recursivo permite eliminar automaticamente os galhos que não possuem dados.

Os resultados, tanto do Apriori quanto do C4.5 foram condizentes com os exemplos utilizados nas seções 2.5.1 e 2.5.2, porém, por executar linhas de comando, além da interface gráfica, utilizou-se o R como ferramenta auxiliar dos dois algoritmos, trocando parâmetros com o sistema criado através de funções da linguagem PHP, específicas para a execução de programas externo. Para o Apriori, devido a problemas enfrentados com a geração de regras dinâmicas, foi utilizado o R para criar as regras. No caso do C4.5, o R foi usado para criar a árvore em forma gráfica, somente para auxílio na visualização, visto que já é gerada uma árvore estruturada, em formato de tabela.

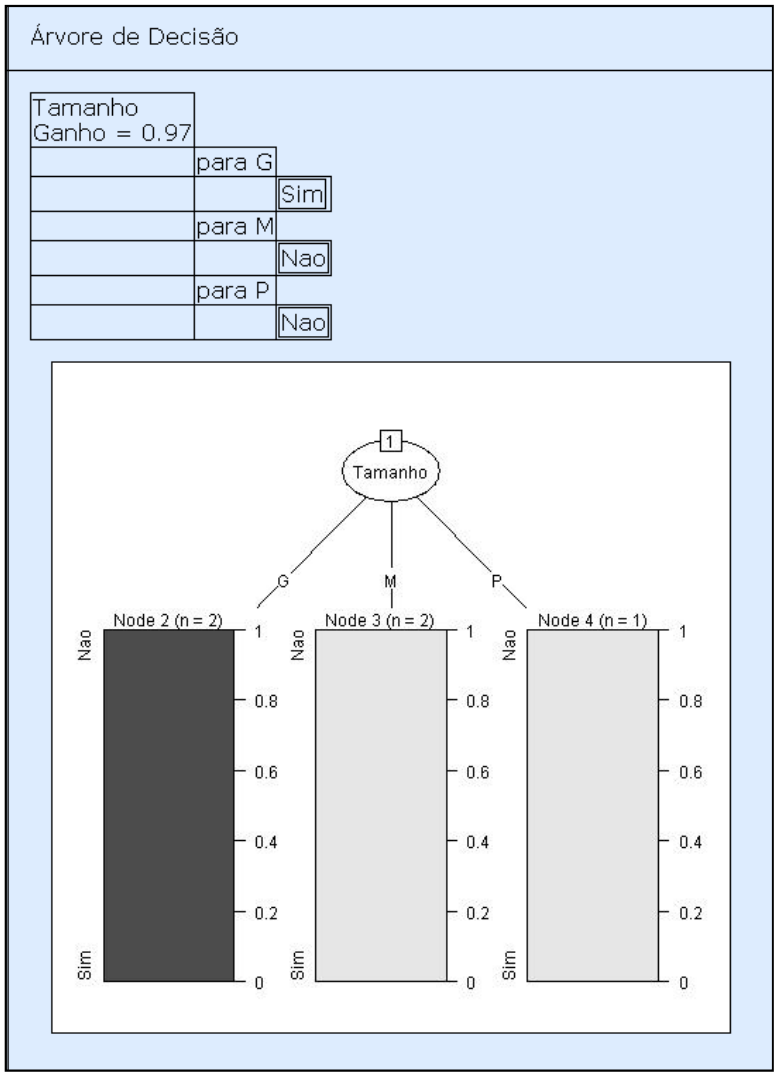


Figura 17 – Árvore de decisão gerada pelo algoritmo C4.5
Fonte: O autor (2012)

4.4 Comparação dos Sistema com as Ferramentas do Quadro 9

Diferente das ferramentas citadas no Quadro 9, o sistema criado apresenta todas as etapas dos algoritmos e todos os cálculos necessários para só então apresentar os resultados.

Por estar programado em PHP, permite-se que o script seja replicado a diversas páginas da internet, pode-se alterar o padrão de entrada, assim como cria a possibilidade de conexão com bancos de dados para armazenamento dos arquivos, de acordo com a necessidade.

Na contagem total dos arquivos que o compõem, o sistema possui 24 KB de tamanho, o que representa apenas 0,88% do menor tamanho apresentado entre as ferramentas.

A opção por código aberto, assim como nas ferramentas citadas, permite a melhoria continua do sistema, assim como a inclusão de novos algoritmos.

5 AVALIAÇÃO DOS RESULTADOS

Considerando que o objetivo principal do presente trabalho deve ser atingido por ambos os algoritmos, pode-se avaliá-los separadamente.

Tanto o C4.5 quanto o Apriori atingiram o objetivo, pois mostram de forma detalhada todo o procedimento além de gerar os resultados do processo. É preciso acrescentar, porém, que a implementação do Apriori possui detalhes a serem observados.

Devido aos problemas em programar todas as combinações possíveis para geração de regras de associação, resultantes do Apriori, não se chegou a um resultado confiável, quando a combinação era maior ou igual a 3x3. A falta de um padrão matemático na forma que os itens se relacionam impossibilitou a codificação da geração de regras, fazendo necessário o uso de uma ferramenta adicional para apresentação dos resultados. O problema do uso de um programa externo, no caso o R, é que, por passar os comandos e variáveis pelo próprio código PHP, cria-se a necessidade da instalação do aplicativo em uma máquina física, o que, hipoteticamente, vai contra os propósitos deste trabalho. No entanto, tendo em vista que a instalação deverá ser realizada apenas uma vez e que não será nas máquinas dos usuários, o sistema se faz válido.

Outro detalhe é que o sistema considerou a linha em branco da tabela como uma variável contendo um espaçamento. Ao tentar adaptar o sistema, deparou-se com o seguinte problema: quando removida a variável, o total das linhas era alterado junto, impactando nos cálculos executados.

Em relação às outras ferramentas, o sistema desenvolvido apresentou resultados satisfatórios. No caso do C4.5, a árvore de decisão gerada em forma de tabela HTML apresentou o mesmo resultado dos quatro softwares (Apendice B) comparados anteriormente:

- para Tamanho G, Compra = Sim;
- para Tamanho M, Compra = Não; e
- para Tamanho P, Compra = Não.

Para uma melhor validação da aplicação do C4.5, fez-se a análise dos dados apresentados na Figura 18, que diferem da tabela utilizada anteriormente por possuir mais registros e valores numéricos.

Aparencia,	Temperatura,	Umidade,	Vento,	Jogo
Sol,	23,	97,	Falso,	Não
Sol,	13,	96,	Verdade,	Não
Encoberto,	17,	98,	Falso,	Sim
Chuvoso,	9,	99,	Falso,	Sim
Chuvoso,	20,	90,	Falso,	Sim
Chuvoso,	12,	91,	Verdade,	Não
Encoberto,	8,	90,	Verdade,	Sim
Sol,	18,	98,	Falso,	Não
Sol,	15,	92,	Falso,	Sim
Chuvoso,	22,	93,	Falso,	Sim
Sol,	21,	91,	Verdade,	Sim
Encoberto,	20,	97,	Verdade,	Sim
Encoberto,	25,	90,	Falso,	Sim
Chuvoso,	19,	100,	Verdade,	Não

Figura 18 – Entrada de dados com valores numéricos – Tabela Jogo
Fonte: O autor (2012)

O sistema desenvolvido gerou uma árvore de decisão tabular, conforme demonstra a Figura 19 abaixo:

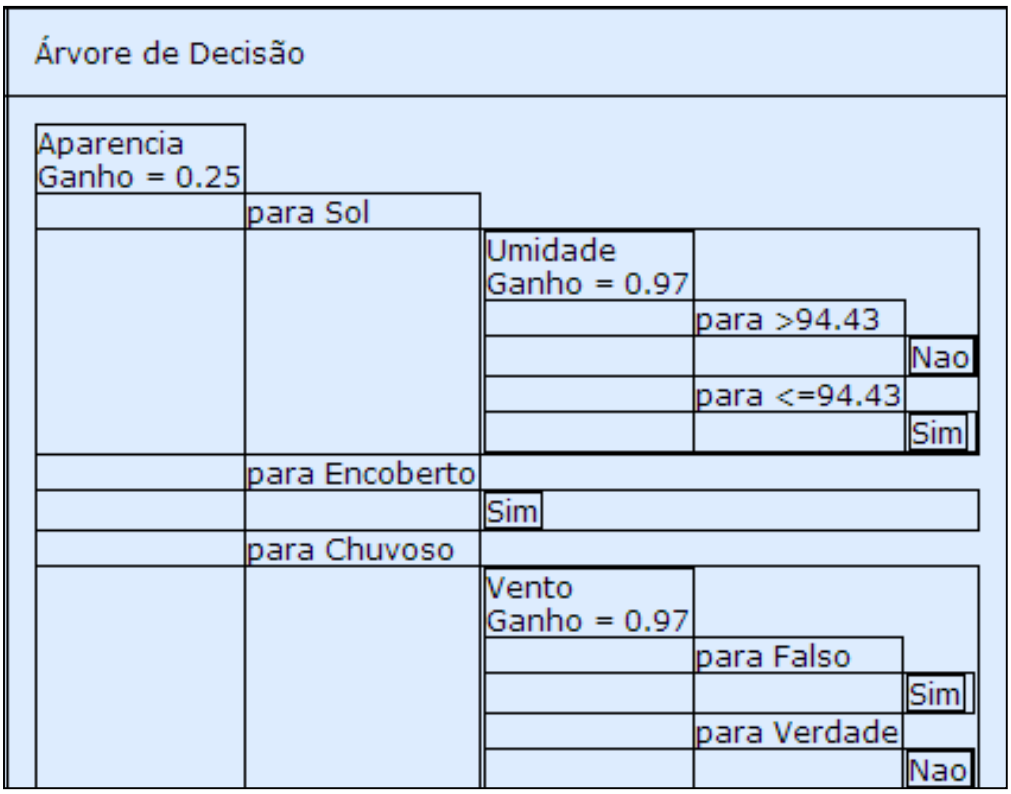


Figura 19 – Resultado do sistema para a mineração da Tabela Jogo
Fonte: O autor (2012)

De forma semelhante, o R e o Weka apresentaram a mesma estrutura de árvore, diferenciando do sistema elaborado somente na discretização do atributo Umidade, onde estes apresentaram o valor 93, conforme Figuras 20 e 21.

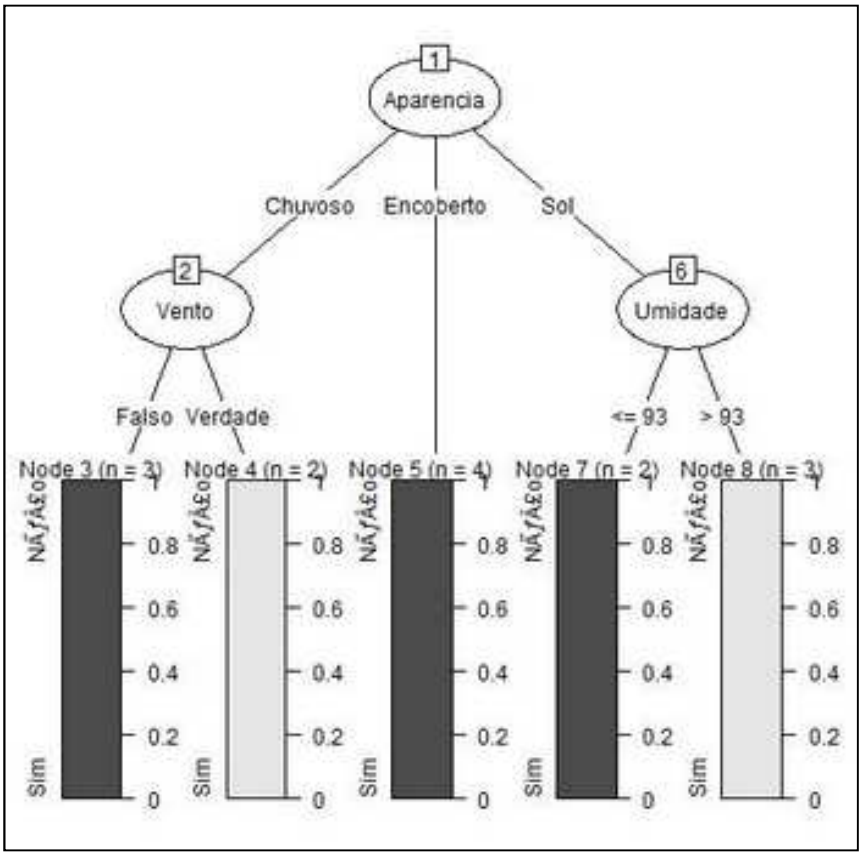


Figura 20 - Resultado do R para a mineração da Tabela Jogo
Fonte: O autor (2012)

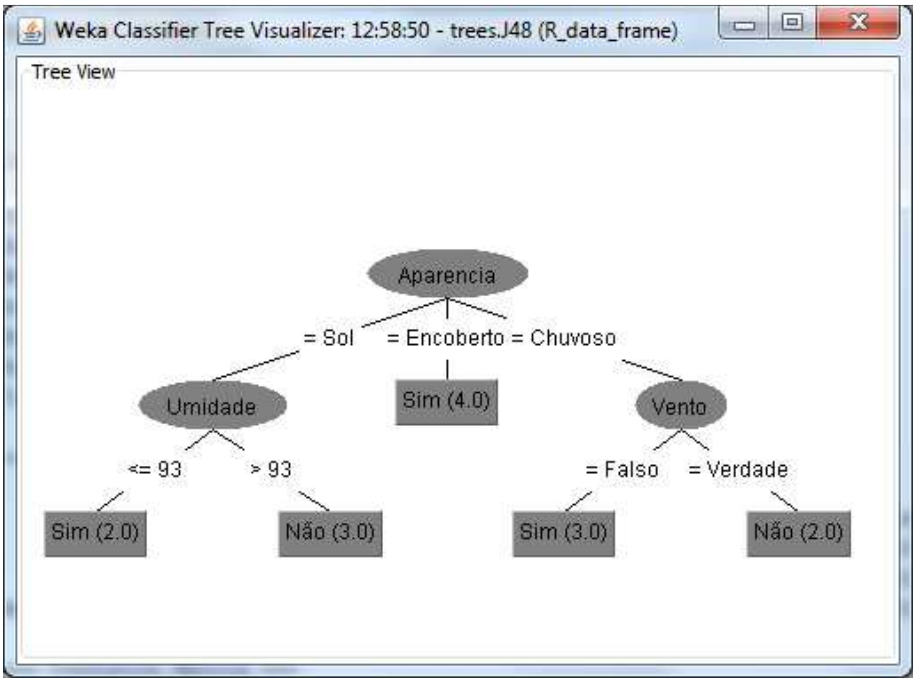


Figura 21– Resultado do Weka para a mineração da Tabela Jogo
Fonte: O autor (2012)

O Tanagra, por sua vez, faz diversas discretizações para o mesmo atributo, como pode ser visto na Figura 22, onde a Temperatura apresenta os valores <22,5000 e <19,5000 como categorias para os valores, o que torna sua árvore de decisão diferente das anteriores.

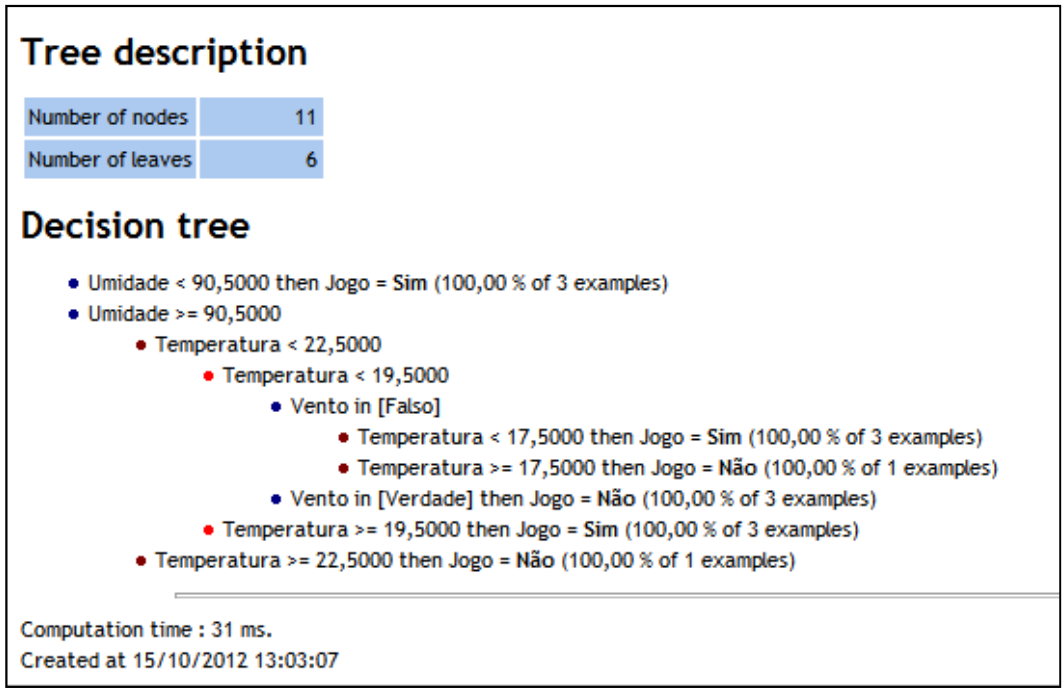


Figura 22– Resultado do Tanagra para a mineração da Tabela Jogo
Fonte: O autor (2012)

O RapidMiner, por não possuir aplicação do algoritmo C4.5 em sua instalação padrão, não permite a mineração da Tabela Jogo.

O resultado do Apriori, quando gerado pelo R, é igual ao resultado gerado pelo Orange, porém difere dos resultados dos outros três softwares (Apendice C) por apresentar, nas regras criadas, somente elementos que ultrapassam o Suporte Mínimo e a Confiança Mínima.

Sob as condições de Suporte Mínimo sendo 40% e Confiança igual a 70%, conforme descrito na seção 2.5.2, tanto o Weka quanto o Tanagra retornaram as mesmas regras, incluindo “Se Batata e Banana então Cenoura” e “Se Tomate então Batata” que não passam pelo Suporte estabelecido. O RapidMiner, por usar em seu sistema o W-Apriori, baseado no Weka, apresenta o mesmo resultado.

6 CONSIDERAÇÕES FINAIS

O crescimento do volume de dados armazenados em meio digital tornou fundamental o tratamento, a seleção e principalmente a análise das informações existentes. Um dos fatores que possibilitaram esse crescimento informacional foi a evolução da tecnologia e o surgimento e expansão da internet. A Mineração de Dados, nesse contexto, se apresenta como uma prática eficaz para examinar os excessos de dados transformando-os em informação útil.

Visto que não há softwares para aplicação online, esta pesquisa objetivou desenvolver uma ferramenta que tivesse como principais características o funcionamento em ambiente *web*, a apresentação do processamento dos algoritmos e que resultasse em informações satisfatórias, quando comparadas com os softwares já existentes. Considera-se assim que este trabalho atingiu todos os objetivos propostos.

Em relação ao primeiro objetivo específico, estudar os principais conceitos relacionados, realizou-se a distinção dos conceitos de dado, informação e conhecimento, para então explicar os tipos de repositórios de dados: banco de dados, *data warehouse*, *data mart* e banco de dados operacionais. Conceituou-se na sequência o *Knowledge Discovery in Databases*, como um processo macro, e a Mineração de Dados, como uma etapa do processo, para então detalhar as aplicações, tarefas, heurísticas e, finalmente, os algoritmos escolhidos: C4.5 e Apriori. Como o intuito era a aplicação em um sistema web, definem-se as linguagens HTML e PHP.

O segundo objetivo foi atingido na seção 3.1, quando são escolhidas as ferramentas de mineração de dados (Weka, RapidMiner, R, Tanagra e Orange) e os critérios para análise dos softwares, listados a seguir:

- tamanho do arquivo de instalação em *quilobytes (KB)*;
- linguagem de desenvolvimento;
- funcionamento em ambiente web;
- possibilidade de uso do algoritmo C4.5;
- possibilidade de uso do algoritmo Apriori;
- suporte a diversos formatos de arquivos de entrada;
- interface em língua portuguesa;

- apresentação do passo-a-passo do processamento;
- inclusão de novos métodos/ operações;
- última atualização no ano corrente.

A partir da análise feita, pôde-se estabelecer as características da ferramenta proposta, definindo, principalmente, o uso das linguagem PHP e HTML, os formatos de entrada padrão *.txt e *.csv, e a apresentação do processamento dos algoritmos.

Após desenvolvido o sistema, utilizou-se um servidor local para simular um ambiente *web* e aplicar testes à ferramenta.

Os resultados do sistema foram analisados na seção 4.3, e comparados aos resultados obtidos pelas outras ferramentas na seção 5, de modo a validar o seu processamento, possibilitando as conclusões pontuadas a seguir.

Embora a ferramenta desenvolvida possua poucas opções, se comparado com outros sistemas, pode-se destacar algumas vantagens no seu uso em relação aos outros aplicativos:

- não precisa ser instalado pelo usuário: reduz tempo de instalação e evita o consumo de memória e espaço em máquina local;
- demonstra os cálculos das técnicas aplicadas: facilita a compreensão da origem dos resultados;
- possui poucos parâmetros de entrada: não requer um vasto conhecimento do usuário;
- apresenta explicação do funcionamento de cada algoritmo: torna o sistema útil para fins didáticos.

As vantagens citadas explicitam que a ferramenta é de grande valia principalmente para estudantes e iniciantes na área da Mineração de Dados. No entanto, por estar situado em ambiente online, possibilita o uso por qualquer interessado, não restringindo a um grupo específico de usuários. Dessa forma, a elaboração desse trabalho se faz uma iniciativa, visto que cria a oportunidade de implementação de outras técnicas de Mineração de Dados ou melhorias nos algoritmos C4.5 e Apriori, com o intuito de explicitar a teoria, aplicando-a no processamento e análise da informação.

REFERÊNCIAS

A HISTÓRIA do PHP Disponível em: <http://br.php.net/manual/pt_BR/history.php.php>. Acesso em: 20 set. 2012.

AGRAWAL, R.; SRIKANT, R. Fast algorithms for mining association rules. In: **International Conference on Very Large Data Bases**, 20., 1994, Santiago de Chile. Proceedings... San Francisco: Morgan Kaufmann, 1994. p.487-499.

AMO, S. A. . Técnicas de Mineração de Dados. In: **Sociedade Brasileira de Computação**, Universidade Federal da Bahia. (Org.). Jornadas de Atualização em Informática. Salvador: Universidade Federal da Bahia, 2004, v. 2, p. 195-236.

ANUPINDI, N. V. **Inmon vs. Kimball** - An Analysis. Disponível em: <http://www.nagesh.com/publications/technology/173-inmon-vs-kimball-an-analysis.html>. Acesso em: 12 nov. 2011.

BOGORNY, V. **Algoritmos e ferramentas de descoberta de conhecimento em bancos de dados geográficos.** 2003. 43 f. Trabalho Acadêmico (Pós-graduação) - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2003. Disponível em: <http://www.inf.ufrgs.br/~vbogorny/ti3-final.pdf>. Acesso em: 27 nov. 2011.

CASTANHEIRA, L. G. **Aplicação de técnicas de mineração de dados em problemas de classificação de padrões.** 2008. 95 f. Dissertação (Pós-graduação) - Universidade Federal de Minas Gerais, Belo Horizonte, 2008. Disponível em: <http://www.cpdee.ufmg.br/~joao/CE/DefinicaoTrabalhoFinal/ProblemaCromatografia/Dissertacao_LucianaCastanheira.pdf>. Acesso em: 27 nov. 2011.

CONSÓRCIO WORLD WIDE WEB (Ed.). **Visão geral do HTML5**. Disponível em: <http://www.w3c.br>. Acesso em: 20 set. 2012.

DAVENPORT, T. H. **Ecologia da informação**: por que só a tecnologia não basta para o sucesso na era da informação. São Paulo: Futura, 1998.

ELMASRI, R E.; NAVATHE, S. B. **Sistema de banco de dados.** 4. ed. São Paulo: Pearson, 2005.

FAYYAD, U. M.; PIATESTKY SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery: an overview. In: FAYYAD, U. M. et al. (Ed.). **Advances knowledge discovery and data mining**. Menlo Park: AAAI, 1996a. p. 1-36. Disponível em: <http://www.kdnuggets.com/gpspubs/aimag-kdd-overview-1996-Fayyad.pdf.> Acesso em 02 out. 2007.

FRAWLEY, W. J.; PIATETSKY-SHAPIRO, G.; MATHEUS, C. J. Knowledge discovery in databases: an overview. In: **AI Magazine**, v. 13, p. 57-70, 1992.

GIL, A. C. **Como elaborar projetos de pesquisa.** 4. ed. São Paulo: Atlas, 2002.

GOEBEL, M.; GRUENWALD, L. A Survey of Data Mining and Knowledge Discovery Software Tools. **ACM SIGKDD Explorations**, New York, v. 1, no. 1, p. 20-33, June. 1999. Disponível em: <http://www.sigkdd.org/explorations/issues/1-1-1999-06/survey.pdf > . Acesso em: 27 nov. 2011.

GOLDSCHMIDT, R.; PASSOS, E. **Data mining**: um guia prático. Rio de Janeiro:Elsevier, 2005.

HAN, J.; KAMBER, M. **Data mining:** concepts and techniques. 2. ed. São Francisco: Morgan Kaufmann Publishers, 2006.

HUSEMANN, B.; LECTENBORGER, J.; VOSSEN G. "**Conceptual *data warehouse* design**," Proceedings of the Intl. Workshop on DMDW 2000, Stockholm, Sweden, Jun, 2000.

INMON, W. H. **Como construir o *data warehouse***. Rio de Janeiro: Campus, 1997. 37p.

_____. ***Data mart* does not equal *data warehouse*.** Disponível em: <http://www.information-management.com/infodirect/19991120/1675-1.html>. Acesso em: 24 maio 2011.

KDNUGGETS. Data Mining Community's Top Resource. **Software suites for data mining, analytics, and knowledge discovery**. Disponível em: <http://www.kdnuggets.com/software/suites.html#free>. Acesso em: 17 out. 2011.

KIMBALL, R. **The *data warehouse* toolkit**: the complete guide to dimensional modeling. New York: John Wiley & Sons, 2002. 436p

_____. ***Data warehouse* toolkit**: Técnicas para Construção de *Data Warehouse*s Dimensionais. São Paulo: Makron Books.1998.

LAKATOS, E. M.; MARCONI, M. A. **Fundamentos da metodologia científica.** 4. ed. São Paulo: Editora Atlas, 2001.

MARTINS, I. M. **A mineração de dados para descoberta de conhecimento e uma oferta adequada no canal de televisão aberta.** 2010. 74 f. Monografia (Graduação) - Universidade Federal do Paraná, Curitiba, 2010. Disponível em: <http://www.decigi.ufpr.br/monografias/2009/IdemaraMarceliMartins.pdf>. Acesso em: 04 dez. 2011.

MIRANDA, R. C. R. O uso da informação na formulação de ações estratégicas pelas empresas. **Ciência da informação**, Brasília, v. 28, n. 3, p.286-292, set./dez. 1999.

MONTEIRO, A. V. G; PINTO, M. P. O.; COSTA, R. M. E. M. Uma aplicação de *Data Warehouse* para apoiar negócios. **Cadernos do Ime**: Série Informática, Rio de

Janeiro, v. 16, n. , p.50-61, jun. 2004. Disponível em: <http://magnum.ime.uerj.br/cadernos/cadinf/vol16/CadernosIME-INF-V16-5-Rosa.PDF> Acesso em:04 dez. 2011.

NARDI, A. R. **Fundamentos e modelagem de bancos de dados multidimensionais.** 2007.Disponível em: <http://msdn.microsoft.com/pt-br/library/cc518031.aspx>. Acesso em: 22 nov. 2011.

NETTO, L. S. **A mineração de dados e o apoio à gestão organizacional.** 2007. 59 f. Monografia (Graduação) - Universidade Federal do Paraná, Curitiba, 2007. Disponível em: <http://www.decigi.ufpr.br/monografias/2007/LeandroNetto.pdf>. Acesso em: 26 nov. 2011.

PACHECO, M. A. C. **Algoritmos genéticos**: Princípios e Aplicações. In: INTERCON99: V Congreso Internacional de Ingeniería Electrónica, 1999, Lima, Peru. Proceedings of the INTERCON99: V Congreso Internacional de Ingeniería Electrónica. Lima, Peru, 1999. p. 11-16.

PIATETSKY-SHAPIRO, G. **In-database data mining advantages/ differences compared to data mining done on a flat file extracted from the database/*data warehouse*?** Disponível em: <http://www.kdnuggets.com/faq/in-database-data-mining.html>. Acesso em: 08 dez. 2011.

QUINLAN, J. R. Induction of decision trees. **Machine learning,** v. 1, n. 1, p. 81-106,1986. ISSN 0885-6125.

REZENDE, S. O. **Mineração de dados**, MiniCurso, XXV Congresso da SBC / ENIA – Encontro Nacional de Inteligência Artificial, São Leopoldo, Brasil, 2005.

SANTOS, I. M. ***Data warehouse* como ferramenta de auxílio em sistemas de monitoramento ambiental.** 2003. 41 f. Trabalho Monográfico (Graduação) - Universidade Federal do Mato Grosso, Cuiabá, 2003.

SETZER, V. W. "Dados, informação, conhecimento e competência". *DataGramaZero.* **Revista de ciência da informação**, n. 0, dez. 1999. Disponível em: <http://www.dgz.org.br>. Acesso em:

TAN, P.; STEINBACH, M.; KUMAR, V. **Introdução ao data mining:**Mineração de Dados. Rio de Janeiro: Ciência Moderna, 2009.

TSUNODA D.Material didático referente à disciplina de Mineração de Dados, no período de 2010/2 do curso de Gestão da Informação da UFPR.

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO. **Questionário para avaliação de softwares**.Disponível em: <http://www.unirio.br/ladoc/disciplinas/files/ibictsoftware.pdf>. Acesso em: 16 dez. 2011.

VENTURA, M. M. O estudo de caso como modalidade de pesquisa. **Revista SOCERJ**.v. 20, n.5 p. 383-386. Set/Out, 2007

WINCKLER, M.; PIMENTA, M. (2002) **Avaliação de usabilidade de sites *web***. In : Nedel, Luciana (Org.) X Escola de Informática da SBC-Sul (ERI2002), Caxias do Sul, Criciúma, Cascavel, Brazil. 2002. p. 85-137.

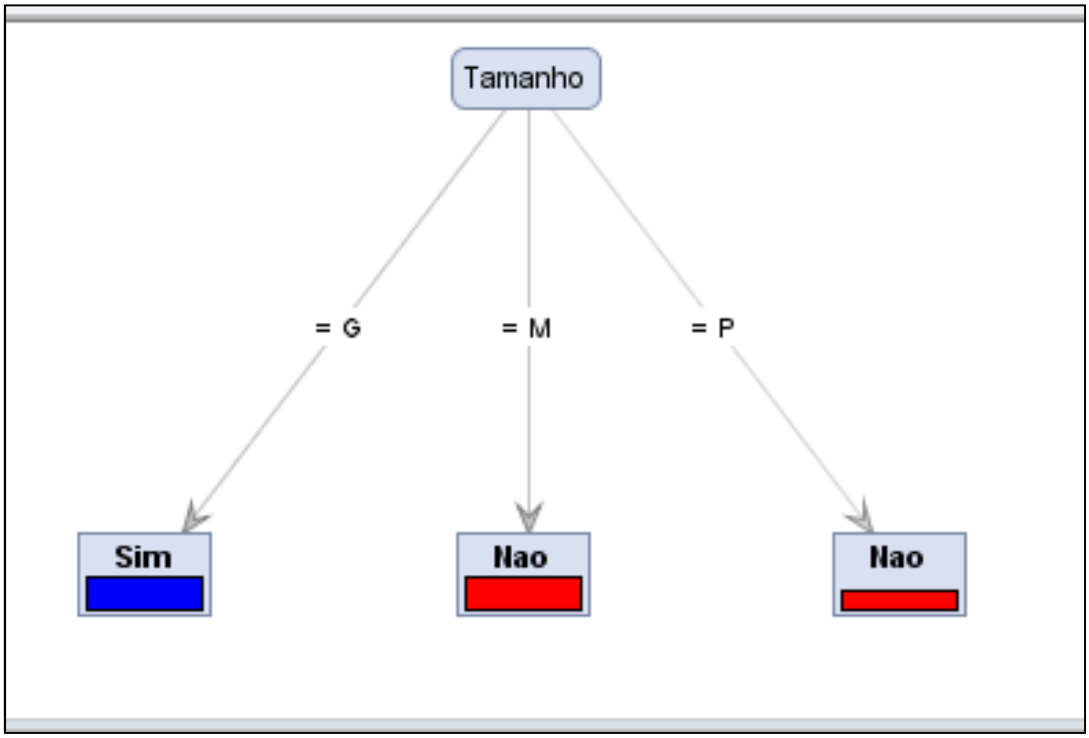
APÊNDICE A – RESULTADOS DOS SOFTWARES PARA O C4.5, TABELA COMPRA CAMISETA

TANAGRA:

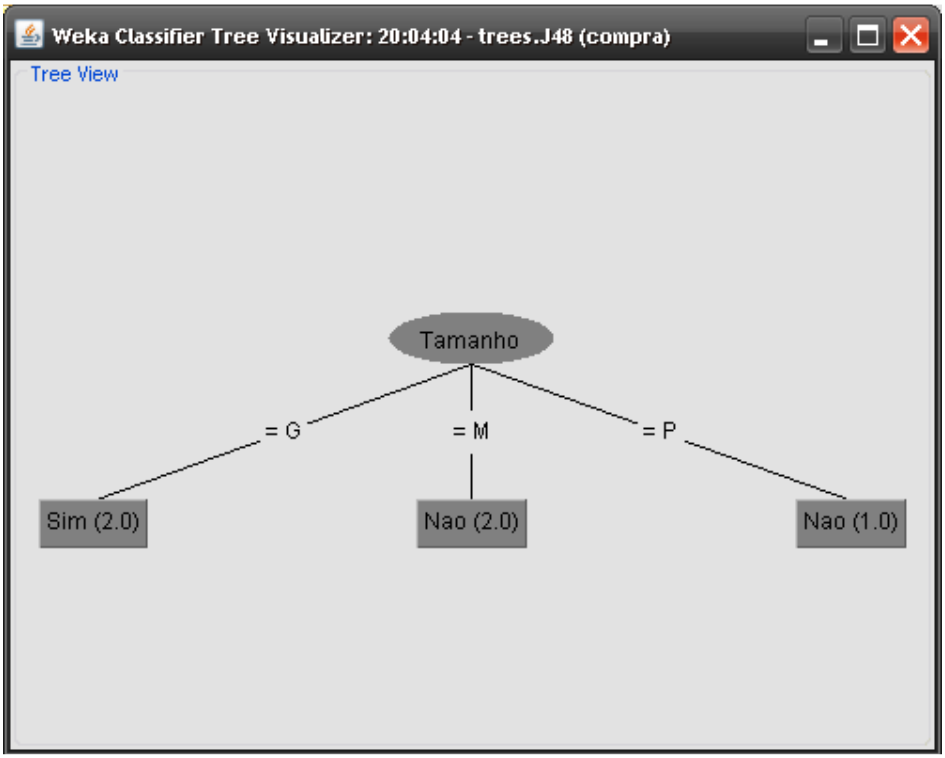
Decision tree

- Tamanho in [G] then Compra = **Sim** (100,00 % of 2 examples)
- Tamanho in [M] then Compra = **Nao** (100,00 % of 2 examples)
- Tamanho in [P] then Compra = **Nao** (100,00 % of 1 examples)

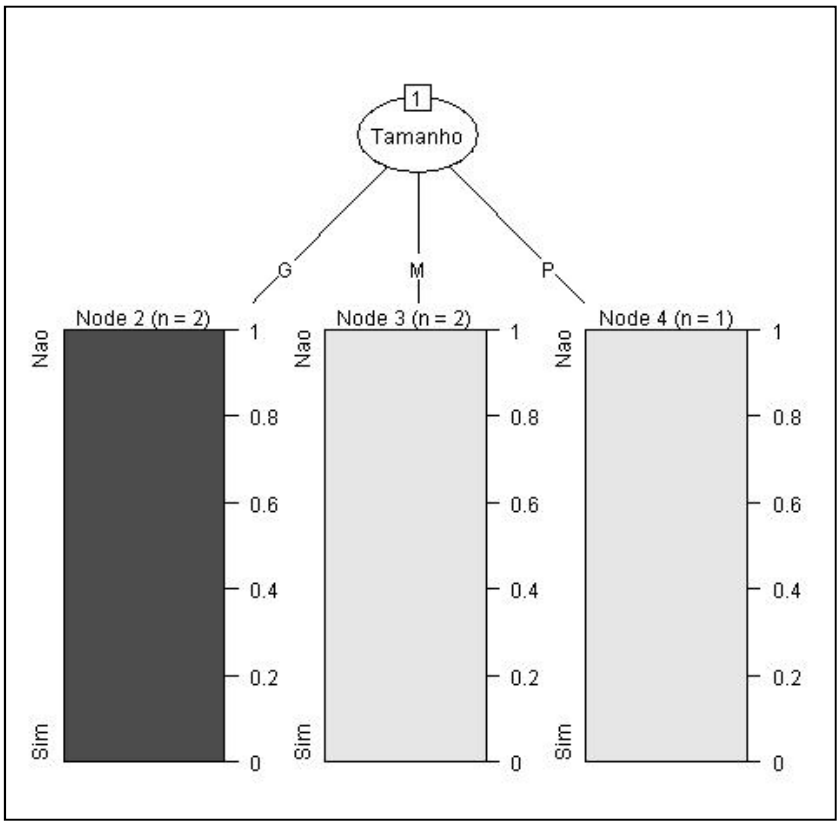
RAPIDMINER:



WEKA:


Weka Classifier Tree Visualizer: 20:04:04 - trees.J48 (compra)
Tree View
Tamanho
= G
= M
= P
Sim (2.0)
Nao (2.0)
Nao (1.0)

R:


1
Tamanho
G
M
P
Node 2 (n = 2)
Node 3 (n = 2)
Node 4 (n = 1)
Nao
Sim
1
0.8
0.6
0.4
0.2
0

APÊNDICE B – RESULTADOS DOS SOFTWARES PARA O APRIORI

TANAGRA:

RULES

Number of rules : 6

N°	Antecedent	Consequent	Lift	Support (%)	Confidence (%)
1	"Cenoura=true"	"Banana=true"	1,50000	50,000	75,000
2	"Banana=true"	"Cenoura=true"	1,50000	50,000	100,000
3	"Batata=true" - "Banana=true"	"Cenoura=true"	1,50000	33,333	100,000
4	"Tomate=true"	"Batata=true"	1,50000	33,333	100,000
5	"Batata=true"	"Cenoura=true"	1,12500	50,000	75,000
6	"Cenoura=true"	"Batata=true"	1,12500	50,000	75,000

WEKA E RAPIDMINER:

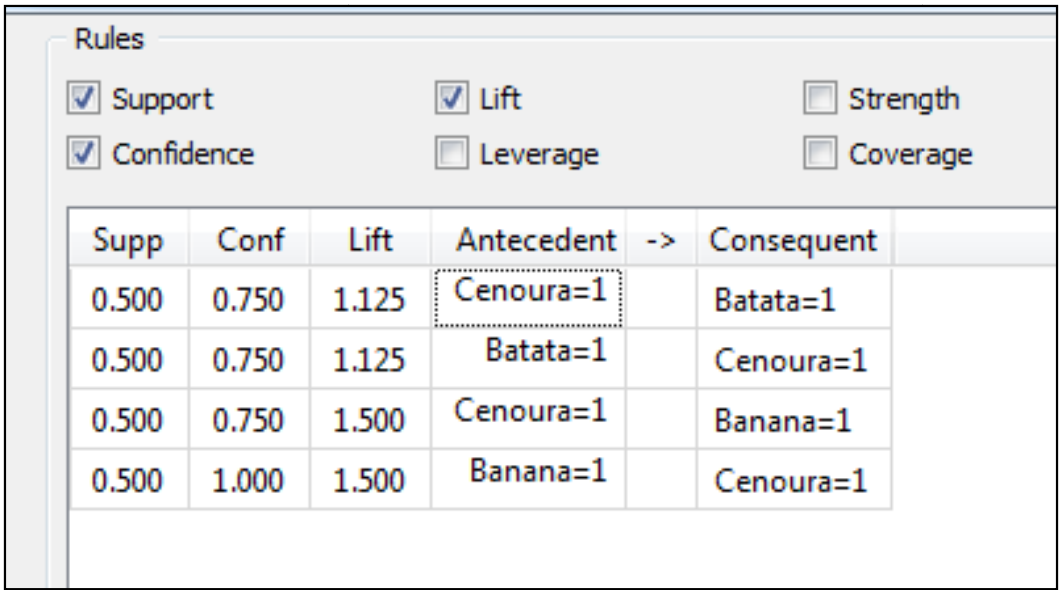
```
Best rules found:

 1. Banana=1 3 ==> Cenoura=1 3    conf:(1)
 2. Tomate=1 2 ==> Batata=1 2    conf:(1)
 3. Batata=1 Banana=1 2 ==> Cenoura=1 2    conf:(1)
 4. Cenoura=1 4 ==> Batata=1 3    conf:(0.75)
 5. Batata=1 4 ==> Cenoura=1 3    conf:(0.75)
 6. Cenoura=1 4 ==> Banana=1 3    conf:(0.75)
```

R:

```
                    rules support confidence  lift
1 {Banana} => {Cenoura}     0.5       1.00 1.500
2 {Cenoura} => {Banana}     0.5       0.75 1.500
3 {Cenoura} => {Batata}     0.5       0.75 1.125
4 {Batata} => {Cenoura}     0.5       0.75 1.125
```

ORANGE:



APÊNDICE C – CÓDIGO FONTE DO PROCESSAMENTO APRIORI

```
<html>
<head>
<link rel="stylesheet" href="style.css" media="screen" />
</head>
<body>


<center>
<?php
error_reporting(E_ALL & ~E_NOTICE & ~E_WARNING);

function comb($a, $len){
    if ($len > count($a))return 'Vazio';
    $out = array();
    if ($len==1) {
            foreach ($a as $v) $out[] = array($v);
            return $out;
    }
    $len--;
    while (count($a) > $len) {
            $b = array_shift($a);
            $c = comb($a, $len);
            foreach ($c as $v){
                    array_unshift($v, $b);
                    $out[] = $v;
            }
    }
          return $out;
}

 function getPermutations ( Array $Array , Array $Permutations = Array ( ) ,
$limitOffset = 10 ) {
          static $permutedItems = Array ( ) ;
          $FlattenArray = Array ( );
          forEach( new RecursiveIteratorIterator ( new RecursiveArrayIterator ( $Array )
) as $Data )
                    $FlattenArray [ ] = $Data ;
                    if( count( $FlattenArray ) > ( intval( $limitOffset ) ) )
                        throw new LengthException ( sprintf( 'NÃ£o podemos gerar
permutaÃ§Ãµes com mais de %d valores' , $limitOffset ) );
          $Array = array_filter ( array_unique ( $FlattenArray ) ) ;
          if( count ( $Array ) ) {
                for ( $i = 0; $i < count ( $Array );  ++ $i ) {
                      $newArray = $Array ;
                      $newPermutations = $Permutations ;
                      array_push ( $newPermutations , array_shift ( array_splice (
$newArray , $i , 1 ) ) ) ;
```

```
                    getPermutations ( $newArray , $newPermutations , $limitOffset ) ;
                }
                return $permutedItems;
            } else $permutedItems [ ] = $Permutations;

    }

$suporte_minimo = $_POST['sup']/100;
$conf_minima = $_POST['conf']/100;

$_FILES['file']['name'];
$_FILES['file']['size'];
$_FILES['file']['type'];
$_FILES['file']['tmp_name'];
$_FILES['file']['error'];

$linha = 0;
$total = array();
echo "<table cellpadding='15'><tr valign='top'><td colspan=3><a
href='javascript:history.go(-1)'>Voltar</a></td></tr><tr valign='top'><td>";
echo "<strong>Dados da tabela:</br>".$_FILES['file']['name']."</strong></br></br>";
echo "<table>";
if (($ledoc = fopen($_FILES['file']['tmp_name'], "r")) !== FALSE) {
    while (($data = fgetcsv($ledoc, 1000, ",")) !== FALSE) {
        $num = count($data);
          echo "<tr>";
        $linha++;
        for ($c=0; $c < $num; $c++) {
            $compras[$linha][$c]=$data[$c];
                        echo "<td>".$compras[$linha][$c]."</td>";
                        array_push($total,$compras[$linha][$c]);
        }
                echo "</tr>";

    }
echo "</table>";

echo "<p><strong>Total de linhas</strong> = ".$total_compras = count($compras);
echo "</br><strong>Suporte M&iacutenimo</strong> = ".$_POST['sup']."%";
echo "</br><strong>Confian&ccedila M&iacutenima</strong> =
".$_POST['conf']."%</p></td><td>";
$suporte = array();
foreach($compras as $row){

    foreach($row as $produto){
        $suporte[$produto]++;
    }
}
echo "<strong>Suporte Individual</strong></br></br>";
```

```
echo
"<table><tr><td><strong>Item</strong></td><td><strong>Suporte</strong></td></tr
>";
foreach($suporte as $key => $row){

          $result = $row/$total_compras;
          echo "<tr><td>".$key."</td><td>".$row."/".$total_compras." =
".round($result,7)."</td></tr>";
    if($row/$total_compras < $suporte_minimo)

        unset($suporte[$key]);
}
echo "</table>";

foreach($suporte as $key => $row){
    $tmp[] = $key;
}
$count_rec = 2;
$int = 1;

while($int != 0){

    $contador = NULL;
    echo "</br><strong>Suporte - Combina&ccedil&atildeo
".$count_rec."x".$count_rec."</strong>:</br></br>";
    $a = comb($tmp,$count_rec);


    foreach($a as $key_com => $row){

        $count_fila = 1;
        foreach($compras as $row1){
            $aaa = array_diff($row,$row1);
            if(count($aaa) == 0)
                $contador[$key_com]++;
        }
    }

    foreach($contador as $key => $row){

        if($row/$total_compras < $suporte_minimo)
            unset($contador[$key]);
    }
echo
"<table><tr><td><strong>Itens</strong></td><td><strong>Suporte</strong></td></tr
>";
          foreach ($a as $key => $linha) {
          echo "<tr><td>"       ;
                    foreach ($linha as $valor){
```

```
                        echo $valor." ";
                        }
        $result = $contador[$key]/$total_compras;
        echo "</td><td>".$contador[$key]."/".$total_compras." =
".round($result,7)."</td></tr>";
      }
echo "</table>";
   if(count($contador) == 0){
      $numero_do_cara = $count_rec - 1;
      $int = 0;
            echo "Vazio";
   }

      $teste[$count_rec] = $contador;

      $count_rec++;

}
echo "</td><td>";
$quase_la = array();

$a = comb($tmp,$numero_do_cara);
foreach($teste[$numero_do_cara] as $key => $row){
   $quase_la[] = $a[$key];
}

echo "<strong>Regras geradas para os itens ";
foreach ($quase_la as $key => $linha) {

            foreach ($linha as $valor){
                  echo $valor." ";
                  }

}
echo ":</strong></br><p>";

$arquivo = "apriori.r";
$fh = fopen($arquivo, 'w') or die("nao criou apriori.r");
$stringData = "library(arules)\n";
fwrite($fh, $stringData);
$_FILES['file']['tmp_name'] = str_replace("\\", "/", $_FILES['file']['tmp_name']);
$stringData = "txn = read.transactions(file='".$_FILES['file']['tmp_name']."',
rm.duplicates= FALSE, format='basket',sep=',',cols =NULL);\n";
fwrite($fh, $stringData);
$stringData = "basket_rules <- apriori(txn,parameter = list(sup = ".$suporte_minimo.",
conf = ".$conf_minima.",target='rules'));\n";
fwrite($fh, $stringData);
$stringData = "sink('apriori.txt')\n";
fwrite($fh, $stringData);
```

```
$stringData = "as(basket_rules, 'data.frame')\n";
fwrite($fh, $stringData);
$stringData = "sink()\n";
fwrite($fh, $stringData);
fclose($fh);

exec("R-2.15.0\\bin\\i386\\Rterm.exe CMD BATCH -q --slave --no-save <apriori.r");

$file = "apriori.txt";
// $en = array("rules", "support", "confidence", "=>");
// $pt = array("regras", "suporte", "confian&ccedila", "entÃ£o");
$f = fopen($file, "r");
while ( $line = fgets($f, 1000) ) {
        if(is_numeric($line)){
        $line = round($line, 2);
        }
$line = str_replace($en, $pt, $line);
print "<pre>".$line."</pre>";
}

echo "</td></tr></table>";
fclose($ledoc);
}
?>
</p>
</center>
</body>
</html>
```

APÊNDICE D – CÓDIGO FONTE DO PROCESSAMENTO C4.5

```
<html>
<head>
<link rel="stylesheet" href="style.css" media="screen" />
</head>
<body>
<center>
<?php

error_reporting(E_PARSE);
// ------------------------------------------------------------
$_FILES['file']['name'];
$_FILES['file']['size'];
$_FILES['file']['type'];
$_FILES['file']['tmp_name'];
$_FILES['file']['error'];

$substituir = array(" ",
"?","ç","Ç","ã","á","à","â","ó","ò","õ","ô","é","è","ê","í","ì","ú","ü","ù","Ã","Á","À","Â","Ó","
Ò","Õ","Ô","É","È","Ê","Í","Ì","Ú","Ü","Ù");
$por = array(".",
".","c","C","a","a","a","a","o","o","o","o","e","e","e","i","i","u","u","u","A","A","A","A","O","
O","O","O","E","E","E","I","I","U","U","U");
$row = 0;
echo "<table cellpadding='15'><tr><td colspan=3><a href='javascript:history.go(-
1)'>Voltar</a></td></tr><tr valign='top'><td>Dados da Tabela de
Entrada:</td><td>Processo</td><td>Árvore de Decisão</td></tr><tr
valign='top'><td>";
if (($ledoc = fopen($_FILES['file']['tmp_name'], "r")) !== FALSE) {
        echo "<table border='1'>";
   while (($data = fgetcsv($ledoc, 1000, ",")) !== FALSE) {
      $num = count($data);
                echo "<tr valign='top'>";
      $linha++;
      for ($c=0; $c < $num; $c++) {
                        $ocorrencia[$linha][$c] = str_replace($substituir, $por, $data[$c]);
                        echo "<td>";
                        if($linha==1){echo "<strong>";}
                        echo $ocorrencia[$linha][$c];
                        if($linha==1){echo "</strong>";}
                        echo "</td>";
                        array_push($total,$ocorrencia[$linha][$c]);
      }
                echo "</tr>";

   }
        echo "</table></br>";
```

```
        $nLinha = $linha-1;
    fclose($ledoc);
}
// ------------------------------------------------------------
// Definição das variáveis principais
// ------------------------------------------------------------
//
// $nLinha = total de linhas da tabela
// $ocorrencia = valores das linhas da tabela
// $atributos = cabeçalho da tabela
// $colunas = valores das colunas das tabelas
// $variaveis = tipos de variaveis de cada coluna
// $totalvariaveis = total de aparições de cada tipo de variavel
//
// ------------------------------------------------------------

        foreach($ocorrencia as $key => $row){

                        if ($key==1){
                        foreach ($row as $m => $n){
                                $atributos[$m] = $n;
                                }
                        }
                        if ($key>1){
                        foreach ($row as $m => $n){
                                $colunas[$m][]=$n;
                                }
                        }

        }

// ------------------------------------------------------------
// Discretização da Tabela
// ------------------------------------------------------------

$contatributos = count($atributos);

foreach ($colunas as $coluna => $linha){
        if(is_numeric($linha[0])){
        $media[$coluna] = array_sum($linha)/$nLinha;
        $att1 = "<=".round($media[$coluna],2);
        $att2 = ">".round($media[$coluna],2);


                foreach ($ocorrencia as $key => $row){
                        if ($key >1){
                                foreach ($row as $c => $valor){
                                if($c==$coluna){
                                        if($valor<=$media[$coluna]){
```

```
                                        $ocorrencia[$key][$c] = $att1;
                                        }
                                else{
                                        $ocorrencia[$key][$c] = $att2;
                                        }
                        }
                        }
                }
        }
        }

}

foreach ($colunas as $key => $row){
        $variaveis[$key] = array_values(array_unique($row));
        $totalvariaveis[$key] = array_count_values($row);
}


        $AM = array_pop($atributos);
        $keyAM = end(array_keys($atributos))+1;

$teste = 0;
foreach ($totalvariaveis[$keyAM] as $key => $row){
        if ($row> $teste){
                $teste = $row;
                $maisocorre = $key;
        }
}
echo "</td><td>";
// ------------------------------------------------------------
// Formula Recursiva para Criação da Árvore
// ------------------------------------------------------------
$level=0;
Processo($ocorrencia, $AM, $atributos, $maisocorre, $level);
function Processo($dataset, $Att_Meta, $Lista_Att, $MO, $nivel){



$Atts = array_values(array_shift($dataset));

        foreach($dataset as $key => $row){

                        foreach ($row as $m => $n){
                                $colunas[$m][]=$n;
                                }
        }

        foreach ($colunas as $key => $row){
```

```
        $variaveis[$key] = array_values(array_unique($row));
        $totalvariaveis[$key] = array_count_values($row);
}


$cabecalho = $Atts;
$AM = $Att_Meta;
$keyAM = end(array_keys($Atts));
$nLinhas = count($dataset);
$contatributos = count($Lista_Att);

$HS=0;
$contador = count($variaveis[$keyAM]);
if($contador!=1){

        echo "<strong>Entropia:</strong></br>H(S) = ";
        for($e=0; $e<$contador; $e++){
                        echo "-
(".$totalvariaveis[$keyAM][$variaveis[$keyAM][$e]]."/".$nLinhas.")log2(".$totalvariavei
s[$keyAM][$variaveis[$keyAM][$e]]."/".$nLinhas.")";
                        $HS = $HS -
($totalvariaveis[$keyAM][$variaveis[$keyAM][$e]]/$nLinhas)*
log($totalvariaveis[$keyAM][$variaveis[$keyAM][$e]]/$nLinhas, 2);
        }
        echo " = <strong>".round($HS,2)."</strong>";
}
else{
        echo "'".$variaveis[$keyAM][0]."' é o unico valor do atributo meta.</br>";
        return;
}

if($contatributos==0){
$teste = 0;

        foreach ($totalvariaveis[$keyAM] as $key => $valor){
                if ($valor > $teste){
                $resultado = $key." = ".$valor;
                }
                else {
                $resultado = $MO;
                }
        }
echo $resultado;
return;
}
echo "</br><strong>Ganho:</strong>";
foreach ($Lista_Att as $key => $value){
 echo "</br>".$value." = ";
        $entropia[$value] = $HS;
```

```
        echo round($HS,2);
        for($i=0; $i<count($variaveis[$key]); $i++){
        ${$i.soma} = 0;
                for($j=0; $j<count($variaveis[$keyAM]); $j++){

                ${$i.$j}=0;
                        foreach ($dataset as $linha => $valores){

                                if(in_array($variaveis[$keyAM][$j] ,$valores) &&
$variaveis[$key][$i]==$colunas[$key][$linha] ) {

                                ${$i.$j}++;

                                }

                        }
                        if(is_infinite(log(${$i.$j}/$totalvariaveis[$key][$variaveis[$key][$i]],
2))){
                                ${$i.$j} = 0;

                        }
                        else{

                                ${$i.$j} = -
(${$i.$j}/$totalvariaveis[$key][$variaveis[$key][$i]])*log(${$i.$j}/$totalvariaveis[$key][$
variaveis[$key][$i]], 2);
                        }
                ${$i.soma} = ${$i.soma} + ${$i.$j};
                }
        echo " -
".$totalvariaveis[$key][$variaveis[$key][$i]]."/".$nLinhas."*(".round(${$i.soma},2).")";
        $entropia[$value] = $entropia[$value] -
($totalvariaveis[$key][$variaveis[$key][$i]]/$nLinhas)*(${$i.soma});
        }
        echo " = <strong>".round($entropia[$value],2)."</strong>";
}
$teste = 0;

        foreach ($entropia as $key => $value){
                if ($value > $teste){
                $raiz['Att'] = $key;
                $raiz['Ganho'] = round($value,2);
                $teste = $value;
                }
        }
echo "</br></br><strong>Raiz = ".$raiz['Att']."</strong></br></br>Para atributo:";

$atributos2 = $Lista_Att;
$nivel = $nivel +1;
```

```
foreach ($Lista_Att as $key => $row){
        if ($raiz['Att'] == $row){
        unset($atributos2[$key]);
        }
}
        foreach ($Lista_Att as $key => $att){
                if ($raiz['Att'] == $att){
                        foreach ($variaveis[$key] as $a => $tipo) {
                        echo "</br>Nivel = ".$nivel.":".$tipo;
                        $subset[$tipo][] = $cabecalho;
                                foreach ($dataset as $linha => $valores){
                                        if ($valores[$key] == $tipo){
                                        array_push($subset[$tipo], $valores);
                                        }
                                }
                                echo "<table border='1'>";
                                foreach ($subset[$tipo] as $p){
                                echo "<tr>";
                                        foreach($p as $o){
                                        echo "<td>".$o."</td>";
                                        }
                                        echo "</tr>";
                                }
                                echo "</table>";
                                if ($subset[$tipo] == $cabecalho){
                                echo $MO;
                                return;
                                }
                                else {

                                Processo($subset[$tipo], $AM, $atributos2, $MO, $nivel);
                                }

                        }

                }
        }

return;
}

function Arvore($dataset, $Att_Meta, $Lista_Att, $MO){



$Atts = array_values(array_shift($dataset));

        foreach($dataset as $key => $row){
```

```
                        foreach ($row as $m => $n){
                                $colunas[$m][]=$n;
                                }
        }

        foreach ($colunas as $key => $row){
        $variaveis[$key] = array_values(array_unique($row));
        $totalvariaveis[$key] = array_count_values($row);
}


$cabecalho = $Atts;
$AM = $Att_Meta;
$keyAM = end(array_keys($Atts));
$nLinhas = count($dataset);
$contatributos = count($Lista_Att);

$HS=0;
$contador = count($variaveis[$keyAM]);
if($contador!=1){
        for($e=0; $e<$contador; $e++){
                        $HS = $HS -
($totalvariaveis[$keyAM][$variaveis[$keyAM][$e]]/$nLinhas)*
log($totalvariaveis[$keyAM][$variaveis[$keyAM][$e]]/$nLinhas, 2);
        }
}
else{
        echo "<table><tr><td>".$variaveis[$keyAM][0]."</td></tr></table></td></tr>";
        return;
}

if($contatributos==0){
$teste = 0;

        foreach ($totalvariaveis[$keyAM] as $key => $valor){
                if ($valor > $teste){
                $resultado = $key." = ".$valor;
                }
                else {
                $resultado = $MO;
                }
        }
echo $resultado;
return;
}
foreach ($Lista_Att as $key => $value){
        $entropia[$value] = $HS;
        for($i=0; $i<count($variaveis[$key]); $i++){
        ${$i.soma} = 0;
```

```
                for($j=0; $j<count($variaveis[$keyAM]); $j++){
                ${$i.$j}=0;
                        foreach ($dataset as $linha => $valores){

                                if(in_array($variaveis[$keyAM][$j] ,$valores) &&
$variaveis[$key][$i]==$colunas[$key][$linha] ) {

                                ${$i.$j}++;

                                }

                        }
                        if(is_infinite(log(${$i.$j}/$totalvariaveis[$key][$variaveis[$key][$i]],
2))){
                                ${$i.$j} = 0;

                        }
                        else{

                                ${$i.$j} = -
(${$i.$j}/$totalvariaveis[$key][$variaveis[$key][$i]])*log(${$i.$j}/$totalvariaveis[$key][$
variaveis[$key][$i]], 2);
                        }
                ${$i.soma} = ${$i.soma} + ${$i.$j};
                }
        $entropia[$value] = $entropia[$value] -
($totalvariaveis[$key][$variaveis[$key][$i]]/$nLinhas)*(${$i.soma});
        }
}
$teste = 0;

        foreach ($entropia as $key => $value){
                if ($value > $teste){
                $raiz['Att'] = $key;
                $raiz['Ganho'] = round($value,2);
                $teste = $value;
                }
        }
echo "<table><tr><td>".$raiz['Att']."</br>Ganho = ".$raiz['Ganho']."</td></tr>";
$atributos2 = $Lista_Att;
foreach ($Lista_Att as $key => $row){
        if ($raiz['Att'] == $row){
        unset($atributos2[$key]);
        }
}
        foreach ($Lista_Att as $key => $att){
                if ($raiz['Att'] == $att){
                        foreach ($variaveis[$key] as $a => $tipo) {
```

```
                        echo "<tr><td></td><td>para
".$tipo."</td></tr><tr><td></td><td></td><td>";
                        $subset[$tipo][] = $cabecalho;
                                foreach ($dataset as $linha => $valores){
                                        if ($valores[$key] == $tipo){
                                        array_push($subset[$tipo], $valores);
                                        }
                                }

                                if ($subset[$tipo] == $cabecalho){
                                echo $MO;
                                return;
                                }
                                else {

                                Arvore($subset[$tipo], $AM, $atributos2, $MO);
                                }

                        }
                        echo "</table>";
                }
        }

return;
}

echo "</td><td>";
Arvore($ocorrencia, $AM, $atributos, $maisocorre);



// -------------------------------------------------------------
//
// -------------------------------------------------------------

$arquivo = "c45.r";
$fh = fopen($arquivo, 'w') or die("nao criou c45.r");
$stringData = "library(RWeka)\n";
fwrite($fh, $stringData);
$stringData = "library(partykit)\n";
fwrite($fh, $stringData);
$stringData = "library(FSelector)\n";
fwrite($fh, $stringData);
$_FILES['file']['tmp_name'] = str_replace("\\", "/", $_FILES['file']['tmp_name']);
$stringData = "b <- read.csv('".$_FILES['file']['tmp_name']."')\n";
fwrite($fh, $stringData);
$stringData = "write.arff(b, 'c45.arff', eol = '\n')\n";
fwrite($fh, $stringData);
$stringData = "m1 <- read.arff('c45.arff')\n";
```

```
fwrite($fh, $stringData);
$stringData = "m2 <- J48(".$AM." ~ ., data = m1)\n";
fwrite($fh, $stringData);
$stringData = "jpeg('arvore.jpg')\n";
fwrite($fh, $stringData);
$stringData = "if(require('party', quietly = TRUE)) plot(as.party(m2))\n";
fwrite($fh, $stringData);
$stringData = "dev.off()\n";
fwrite($fh, $stringData);
fclose($fh);

exec("R-2.15.0\\bin\\i386\\Rterm.exe CMD BATCH -q --slave --no-save <c45.r");

echo "<p><img src='arvore.jpg'></p>";

        echo "</td></tr></table>";
// ------------------------------------------------------------
// Apaga a tabela
// ------------------------------------------------------------


?>
</center>
</body>
</html>
```